

Movescout
powered by Suunto
com



SUUNTO 應用程式

編寫指南 - MOVESCOUT.COM



內容

INTRODUCTION	9
簡介	11
APP EXAMPLES 應用程式範例	12
SPRING COUNTER FOR CYCLING 自行車衝刺次數計算器	12
COOPER TEST – WITH SMART ESTIMATION COOPER 測試 – 智慧估算模式	15
APP BASICS APP 基本知識	21
LIFECYCLE OF AN APP App 的生命週期	27
APP DESIGNER App 設計平台	33
APP ZONE APP 專區	35
USING APP IN DEVICE 在 Suunto 裝置上使用 App	36
LOGGING APP RESULTS 紀錄 App 結果	38
APP USER AGREEMENTS APP 使用者協議	41
APP CODE REFERENCE APP 說明代碼參考	42
RESULT 結果	42

PREFIX AND POSTFIX	前綴字與後綴字	43
CASE SENSITIVITY	使用大寫字母	46
SEMICOLON	以分號結尾	47
ERRORS	錯誤	48
COMMENTS	註解	51
MATH FUNCTIONS	數學函數	52
TRIGONOMETRIC FUNCTIONS	三角函數	52
1.	SIN(X)	52
2.	COS(X)	53
3.	ATAN2(Y,X)	53
4.	TAN(X)	54
5.	SIND(X)	55
6.	COSD(X)	56
7.	TAND(X)	56
8.	ATAND2(Y,X)	57
OTHER MATHEMATICAL FUNCTIONS	其他數學函數	58

1. SQRT(X)	58
2. EXP(X)	58
3. LOG(X)	59
4. LOG10(X)	60
5. POW(X,Y)	60
6. HYPOT(X,Y)	61
7. RAND()	62
8. MOD(X,Y)	63
9. ABS(X)	63
10. ROUND(X)	64
11. CEIL(X)	65
12. FLOOR(X)	65
SUUNTO FUNCTIONS SUUNTO 函數	67
1. ALARMBEEP()	67
2. LIGHT()	67
3. DISTANCE(LAT, LON)	67

4. DISTANCE(LAT1, LON1, LAT2, LON2)	68
5. HEADING(LAT, LON)	69
6. HEADING(LAT1, LON1, LAT2, LON2)	70
OPERATORS AND STRUCTURE	71
RELATIONAL OPERATORS	72
LOGICAL OPERATORS	72
LOGICAL STRUCTURES	74
VARIABLES	77
OWN VARIABLES	77
DIFFERENT WATCH VARIABLE TYPES	79
REAL TIME VALUES	79
LAP VALUES	79
PREVIOUS LAP VALUE	80
INSTANT VALUES	80
TAIL VALUE	80
WATCH VARIABLES	84

SPEED	84
DISTANCE/GPS	86
HR	88
ALTITUDE	90
ENVIRONMENT	93
TIME	95
SWIMMING	96
POWER	98
CADENCE	99
PERSONAL	100
NOT SUPPORTED FEATURES 不支援的功能	103

INTRODUCTION

You have probably done your fair share of sports if you are reading this. You have likely spent loads of your time measuring, analyzing and even seeking guidance in the pursuit of hard facts and a deeper understanding of your performance.

And yet you are left wanting. Perhaps you are still missing that one crucial fact that will help you go harder, faster, longer. We want to change that.

Every one of us has our own unique ambitions. Each sport, each person, each session can have a different set of targets.

We at Suunto have long understood this, and we also understand that sports enthusiasts now their stuff. You know what you really need. Even better, your knowledge could lead to the ultimate tools for like-minded athletes.

With Suunto Apps, we strive to make all that possible. We want you to be able to get the most out of your activities. We want you to innovate, create and change the way sports devices are used. And we want to connect you, with your personal experiences and expertise, to the larger sports community, so that people can also learn to excel.

So, are you ready?

Take the plunge and become a Suunto App developer with the help of this document. We have tried to cover all the key areas you need to know to create the best App you can. If you have any questions, comments or development ideas, please, let us know by sending us your feedback at [Movescount.com](https://movescount.com).

簡介

正在閱讀這篇簡介的你可能已經完成了『份內』的運動量，並且花了許多時間測量、分析、甚至想找專人幫忙解讀你的運動報告上面那些看來艱深難懂的數據。

但是你仍然感到意猶未盡，彷彿就是還缺了那片能讓你更努力、跑更快、動更久的關鍵的拼圖 – 這正是現在我們要改變的。

我們每個人都有各自的企圖心；不同的運動、不同的人、不同的階段都該設定不同的目標。

Suunto 非常明白，我們也清楚只有運動愛好者心理一清二楚：自己最知道自己在追求什麼。甚至，你的知識能打造終極的工具，與你志同道合的運動員都能受益。

透過 **Suunto Apps**，我們努力使這一切成為可能：我們希望激勵你做到最好，希望你發揮想像力、顛覆運動裝置的使用方法，更希望透過你自己的經驗與專業、將你與運動社群連結在一起，與其他成員一同切磋、一同成長。

所以，你準備好了嗎？

我們盡力使這本小冊子涵蓋所有你需要知道的關鍵領域，好幫助你變身為 **Suunto App** 的開發者。若有任何問題、建議、或是程式開發上的好主意，歡迎你在 [Movescount.com](https://www.movescount.com) 回饋給我們。

APP EXAMPLES 應用程式範例

SPRING COUNTER FOR CYCLING 自行車衝刺次數計算器

For cyclists it might be interesting to get a count of how many sprints were done during the workout. This can be calculated with the following App.

自行車運動員可能很有興趣知道自己在運動過程中做了多少次衝刺。要計算這個數據，可以利用以下的程式：

First create 2 variables: **Counter = 0** and **Counted = 0**.
首先創建兩個變數：**Counter = 0** 與 **Counted = 0**.

```
/* While in sport mode do this once per second */  
RESULT=Counter;  
if (SUUNTO_BIKE_POWER_AVG[3] > 300 && Counted ==  
    0){ Counter = Counter + 1;  
RESULT = Counter;  
Counted=1;  
}  
if (SUUNTO_BIKE_POWER_AVG[3] <=  
    300 ){ Counted=0;  
}
```

The logical assumption is that each time the cyclist goes over a power level of 300W and maintains that for at least 3 seconds, it is considered a sprint. When this has been counted, the same intensive period is not counted again as a new sprint the 3-second average power level drops below 300W. So if you do hard interval sessions, this App will show for example "Peaks 50," the amount of times when you have been able to keep your power above 300W for at least 3 seconds.

邏輯上的假設是：每當自行車運動員功率超過 300W、並持續 3 秒以上，就視為是一次衝刺。當上述強度的動作被計為一次衝刺之後，後續相同強度的動作就不會再被另外計算，直到前述的 3 秒平均功率降到 300W 以下。所以如果你真的很努力衝刺，這個應用程式可能會顯示 "Peaks 50" 這樣的數字，顯示你在該次運動中，將 3 秒平均功率維持在高於 300W 的水平次數。

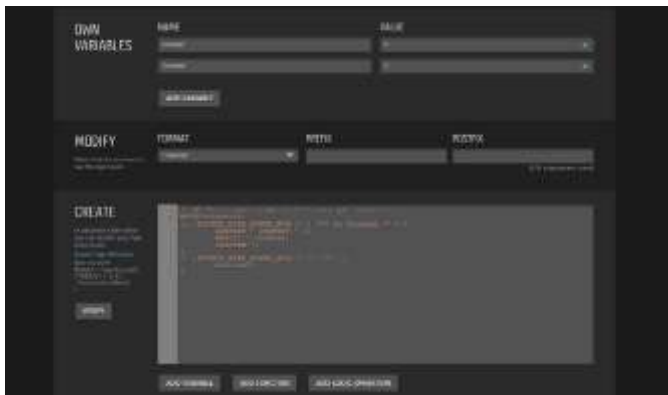


Figure 1 - Example of cycling sprint counting based on watts, buffered value, if-logic, own variables

圖 1 - 自行車衝刺次數計算程式範例，使用瓦特、緩衝值、邏輯計算子 if、自有變數。

COOPER TEST - WITH SMART ESTIMATION COOPER 測試 - 智慧 估算模式

This [App](#) shows you the estimation for finish distance until you reach 12 minutes. Estimation is based on your progress, average pace and recent speed. This gives stable estimation, which is reactive to speed changes and time left. After 12 minutes it BEEPS and shows the end results.

這個 [App](#) 能即時顯示預估完成的距離，直至你完成 12 分鐘的跑步時程。它依據你跑步的進程、平均步速、現時速度來計算預估完成的距離，因為它的計算能依速度變化與剩餘時間調整，所以能提供穩定不失真的預測。12 分鐘後發出蜂鳴聲，並顯示最終的結果。

Create 6 variables: **raceTime = 720, timeLeft = 0, progress = 0, left = 0, Cooper = 0** and **forecastSpeed = 0**

創建六個變數: **raceTime = 720, timeLeft = 0, progress = 0, left = 0, Cooper = 0** 與 **forecastSpeed = 0**

```
timeLeft = raceTime - SUUNTO_DURATION;  
progress = SUUNTO_DURATION / raceTime;  
left = 1-progress;
```

```
forecastSpeed = (SUUNTO_SPEED_AVG[30] * progress) +  
(SUUNTO_AVG_SPD * left);
```

```
if (raceTime ==  
SUUNTO_DURATION){ Cooper =  
SUUNTO_DISTANCE*1000;  
Suunto.alarmBeep();  
}
```

```
if (raceTime > SUUNTO_DURATION){  
RESULT = SUUNTO_DISTANCE * 1000 + ((forecastSpeed/3.6)*timeLeft);  
}else {  
RESULT = Cooper;  
}
```



Figure 2 - Estimation App with various own variables

圖 2 - 依據各種自有變量的作預估計算的 App

RACE TIME PREDICTOR 5K – MARATHON 完賽時間預估程式：5K 馬拉松

Sometimes you need to show more than one info on screen inside on [App](#). Here's an implementation for that.

開啟 [App](#) 時，若需要在螢幕上顯示多個訊息，可利用以下的程式：

First create 3 variables: **print = 0**, **myCounter = 0** and **estimateScreen = 0**.

首先創建三個變數：**print = 0**, **myCounter = 0** 與 **estimateScreen = 0**.

```
/* While in sport mode do this once per second*/  
/* Make user run 1 km before any estimation is given, show the duration to  
estimate*/  
if((SUUNTO_DISTANCE < 1) && (estimateScreen == 0)){  
    print = ((1-SUUNTO_DISTANCE)*1000)/(SUUNTO_AVG_SPD*1000/3600);  
    prefix = "Run";  
    postfix = "h";  
}
```



```
if ((SUUNTO_DISTANCE >= 1) && (estimateScreen == 0)){  
    estimateScreen = 1;  
    myCounter = 0;  
}
```

```
/*Now when user have ran more than 1km*/  
/*Scroll through the different time estimation screens*/  
if ((estimateScreen == 1) && (myCounter >= 2)){  
    print = SUUNTO_DURATION *  
    Suunto.pow((5000/(SUUNTO_DISTANCE*1000)), 1.06);  
    prefix = "5K"; myCounter = 0; estimateScreen = 2;  
}  
if ((estimateScreen == 2) && (myCounter >=2)){  
    print = SUUNTO_DURATION *  
    Suunto.pow((10000/(SUUNTO_DISTANCE*1000)), 1.06);  
    prefix = "10K"; myCounter = 0; estimateScreen = 3;  
}  
if ((estimateScreen == 3) && (myCounter >=2)){  
    print = SUUNTO_DURATION *  
    Suunto.pow((21097/(SUUNTO_DISTANCE*1000)), 1.06);  
    prefix = "1/2 M"; myCounter = 0; estimateScreen = 4;  
}  
if ((estimateScreen == 4) && (myCounter >=2)){  
    print = SUUNTO_DURATION *  
    Suunto.pow((42195/(SUUNTO_DISTANCE*1000)), 1.06);
```

```
prefix = "Mara"; myCounter = 0; estimateScreen = 1;  
}
```

```
myCounter = myCounter+1; RESULT = print;
```

NOTE: If you copy the code to code editor, please fix the quotation marks.

注意：將上述程式直接輸入程式編輯器時，請記得將引號移除。

APP BASICS APP 基本元素

Each App is described with the following elements.

每個 App 都以下述的元素加以描述。

App image: An image is used to showcase the App in App Zone and in your own App library. The image should reflect the nature and content of the App you have created. The text “APP” is automatically overlaid on top of the image.

App 圖像: 在 App Zone 與你自己的 App Library 中，用圖像方式展示 App。圖像應反映出你所建立的 App 的性質與內容。“APP”的字樣會自動覆寫在圖像上。

App name: The App name should be descriptive. However, the App name is also used in URL, the shorter the name is, the better.

App 名稱: App 名稱應具描述性。然而，由於 App 的名稱也會用於超連結中，所以名稱越短越好。

Activity: Choose the main sport for the App. This helps other users find your App.

活動：為 App 選擇適用的主要運動，有助於其他使用者找到你的 App。

Publicity: To allow your App to be used by others, select “public.” Otherwise, select “private.”

公開狀態：若要允許其他人使用你的 App，選擇 “public (公開)”；否則請選擇 “private (私有)”

Category: Select the category that best fits your App. This category is used to filter search results.

分類：選擇最適合你的 App 的分類，有助於過濾搜索結果。

Description: The description is very important for others to understand the purpose of your App. Be thorough and use concrete examples whenever possible.

說明：說明是非常重要的，能夠讓別人了解你的 App 的目的。儘可能地完整並使用具體例子。

Tags: Add tags for your App to help others find it in the App Zone.

標籤：為您的 App 添加標籤，以幫助他人在 App 專區中找到它。

Website: Use the link field to provide additional information about the App. For example, you can link to research that the App is based on or a website of an event where the App would work perfectly.

網站：使用連結欄位提供關於 App 的附加訊息。舉例來說，你可以連結到 App 編寫的研究基礎網站，或是 App 可以發揮最佳功效的事件。

Compatible devices: Devices where this App can be used are listed here.

適用裝置：App 適用的裝置列在此處。

Shoutbox: Every public App has a shoutbox. This enables potential users to ask questions directly from the developer or from other App users.

線上留言板：每個公開的 App 都有線上留言板，讓潛在使用者可以直接向開發者或其他 App 使用者提問。

Thumbs up: Each App can be “liked” by pressing the thumbs up icon. Popularity can be used to sort Apps in App Zone.

按讚：使用者可以對每個 App 按讚。受歡迎程度可以用來排序 App 專區裡面的 App。

Simulator: Each App can be simulated in App Designer. The simulator shows what the App result would be in different situations.

模擬器：每個 App 可以在 App 設計平台中進行模擬。模擬器能夠展示在不同情況下 App 的運算結果。

Unique URL: Each App has a unique URL that can be used on websites, blogs and forums to link to directly to the App page. The address of the app is *www.movescount.com/apps/xxxxID-Appname* where XXXID is the unique ID for the App and APPNAME is the name of the App ad defined by the developer.

專有的超連結：每個 App 都有一個專有的超連結，可以嵌入在網頁、部落格及論壇，讓其他使用者直接連結到 App 的網頁。App 專有的超連結如下：*www.movescount.com/apps/xxxxID-Appname*，其中 XXXID 是 App 獨特的 ID 碼，APPNAME 則是程式開發者為 App 取的名稱。

App creator: App creator shows the name of the person who created the App.

App 建立者：顯示 App 編寫人的姓名。

App create date: Each App shows the date created.

App 建立日期：顯示 App 建立的日期。

TIP! Share your Apps URL and tell others about your latest App.

小撇步! 分享你的 App 連結，並告知大家你最新開發的 App。

LIFECYCLE OF AN APP App 的生命週期

When an App is created, it can have different states within Movescount.com.

當一個 App 被編寫出來，它在 Movescount.com 上會有不同的狀態。

Creation of the App: Anyone who is registered in Movescount.com can use App Designer to create new Apps. This does not require ownership of an App-compatible device. For example, if you want to create an App for your friend who has such a device, you can do so without owning the device yourself.

新建的 App：任何一位 Movescount.com 的註冊會員都可以使用 App 設計平台編寫新的 App；編寫者並不一定需要擁有 App 相容的 Suunto 裝置。例如，你想為擁有 Suunto 裝置的友人編寫一個 App – 在這個情況下，你可以著手編寫 App 而不需要持有 App 相容的 Suunto 裝置。

Private Apps: Only the Movescount.com member who created a private App can utilize it. Private Apps are not viewable at all in App Zone.

私有的 App：只有新建這個 App 的 Movescount.com 的註冊會員才能夠使用它；私有 App 在 App Zone 裡是看不到的。

TIP! When you are creating and testing a new App, it is good to keep it private until you have tested it well enough with your watch. At the moment it's not possible to edit Apps that are used by other members.

小撇步! 新 App 在編寫與測試階段比較適合設為私有狀態。當你用你自己的 Suunto 裝置充分測試之後，就可以將 App 公開- 此時就不能再對 App 進行編輯，因為可能有其他人正在使用它。

Public Apps: Public Apps are shown in App Zone. Any registered member can download and use the App. If an App from App Zone is stored in a member's own library, the App cannot be deleted from Movescount.com.

公開的 App：公開的 App 會顯示在 App Zone 中，任何一位會員都可以下載它來使用。若 App Zone 中的 App 被儲存在某位會員自己的 Library 中，這款 App 就不能從 Movescount.com 被移除。

App in device: An App stored in a member's library can be used in any compatible device. Compatibility is device specific, so not all the Apps can be used with a given device. Apps are added or removed from the device by customizing sports modes (Gear – Customization – sport modes).

裝置內的 App：儲存在會員 Library 中的 App 可以被下載到相容的 Suunto 裝置中使用；相容性是因裝置而異的，所以沒有任何一款裝置能夠使用全部的 App。使用者可以透過自訂運動模式來新增/移除 App (Gear – Customization – sport modes)

Logged Apps: When an App is taken into use, the App results are logged according to the device- specific logging functionality. Logged results are shown as graphs for the recorded Move when uploaded to Movescount.com.

產生紀錄的 App：當開始被使用 App 的時候，產生的資料都會透過各個裝置內建的日誌紀錄功能記錄下來。

Apps in library: The App library is your personal storage area for Apps. All the Apps you have created are automatically stored in your own App library. Public Apps in App Zone can be saved by Movescount members in their own App libraries.

收藏的 App : App Library 是會員儲存 App 的個人區域。你所編寫的 App 會自動儲存在你個人的 App Library ; Movescount 會員則可以自由將 App Zone 裡面已經公開的 App 加到自己的 Library 中。

Removing logged App data from Move: Logged App data cannot be deleted from a recorded Move without deleting the whole Move from Movescount.com.

從 Move 紀錄移除 App 數據 : App 記錄的數據不能單獨從 Move 紀錄中刪除 ; 只能從 Movescount.com 刪除整個 Move 紀錄。

Deleting App from Movescount.com: An App can be deleted from Movescount.com only if no other member has it in his or her App library, the App is not in use by a device (not in any user's sport mode) and logged results from the App are not stored in any Moves. An App remains in

Movescount.com as long as there is at least one person using the App.

從 Movescount.com 刪除 App：只要 App 還符合下列條件，就不能從 Movescount.com 刪除它，1) 沒有任何一位會員將它儲存在自己的 Library、2) 沒有任何一架裝置在使用它（不存在於任何一位使用者的運動模式中）、3) App 記錄的資料也沒有儲存在任何 Move 上。簡言之，只要還有一個人還繼續使用它，這款 App 就會繼續留存在 Movescount.com。

Removing App from App Zone: When the create of a given App creator deletes the App from his or her own App library, the App is also removed from the App Zone. Other members cannot take the App into use anymore. However, members who have earlier saved the App to their App library will still be able to use it.

從 App Zone 移除 App：當一款 App 的編寫人將該款 App 從自己的 Library 移除之後，該款 App 也同時從 App Zone 中被移除。因此，其他的會員就無法再使用該款 App；但是已經將該款 App 收藏到自己 Library 的會員仍然可以繼續使用。

Public App to Private App: If App was public originally and then changed to private, it will not be shown in App Zone anymore. However, members who have earlier saved the App to their App library will still be able to use it.

將公開 App 改為私有：原本已公開的 App 若改為私有狀態，則無法再顯示在 App Zone 中。但是已經將該款 App 收藏到自己 Library 的會員仍然可以繼續使用。

APP DESIGNER App 設計平台

You can design Apps in two different modes.

您可以在兩種模式下編寫 App :

- **The graphic mode** offers the possibility to easily create mathematical equations with different variables, math operators (+, -, x, /) and structures (parentheses) without any coding.
- **圖形模式**讓您使用各種變數、數學運算子（加減乘除）與括弧，簡單建立數學公式，而不需要另外學習程式編寫。
- The **advanced mode** offers the possibility to do more complex equations with various logical operators and functions.
- **進階模式**則提供您多樣的邏輯運算子與函數，讓您可以編輯更為複雜的方程式。



Figure 3 - App Designer: Graphic mode and advanced mode

圖 3 – App 設計平台：圖形模式與進階模式

APP ZONE APP 專區

App Zone can be found at www.movescount.com/Apps.

From App Zone you can find all public Apps made by other Movescount.com members.

App Zone (www.movescount.com/Apps) 收錄了所有 Movescount.com 的註冊會員所編寫並公開的 App。

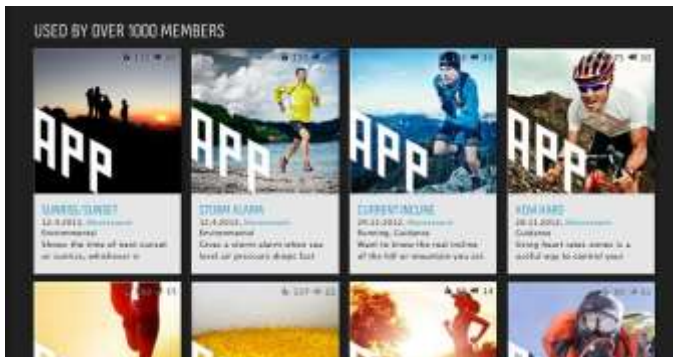


Figure 4 - App Zone in Movescount.com

圖 4 -- Movescount 運動平台上的 App 專區

USING APP IN DEVICE 在 Suunto 裝置上使用 App

Clicking on an App in the App Zone takes you to the home page for that App. There you can find a list of compatible devices for the App. Click the “Save App” button to take the App into use. Any App you save is stored in your personal App library. Once an App is in your library, you can select the App when customizing your devices sports modes.

在 App Zone 點擊任一個 App，會帶您到該款 App 的主頁。在 App 的主頁，您可以找到適用該款 App 的設備列表。點擊“Save App (儲存 App)” 按鈕，便可將 App 下載使用。您所儲存的 App 都會收藏在您個人的 App Library。要使用該款 App 時，只需在您的 Suunto 裝置上自訂運動模式並選取 App 即可。



Figure 5 - Selecting App into device in Sport mode customization

圖 5 - 自訂運動模式、選取並使用 App

LOGGING APP RESULTS 紀錄 App 結果

Apps can be logged as part of a Move recording. Logging App results can be automatic or modified in device settings. App 的數據可以作為 Move 的一部分記錄下來；在裝置設定中可以決定自動紀錄或修改 App 數據。



Figure 6 - Logging options for App in Sport mode customization

圖 6 – 自訂運動模式中，App 的數據紀錄選項

When the App results are logged, you analyze the data once you have uploaded Move to Movescount.com.

App 數據紀錄完成後，您可以將 Move 上傳至 Movescount.com，並在平台上加以分析。

The example below shows the results of an App that calculates running efficiency while running. The App name is shown below the graph as one of the view options. The Y-axis has the prefix "Effic." and when an individual data point is highlighted, you can see the App details over the graph.

下面的範例顯示一款計算跑步效率的 App：App 的名稱顯示在圖表下方(可自訂)、Y 軸"Effic."表示跑步效率，點到任一個單一的資料點時，詳細的資料就會出現在旁邊。

The logged App data is shown in the same way as any other data that Move has. In an App with GPS data record, color coding indicates the App data on the track, and you can select any portion of the graph to see the data distribution as a bar chart as well as a curve.

記錄下來的 App 數據與 Move 其他數據的顯示方式相同。數據包含 GPS 資料時，App 數據會以顏色標示於路徑上，您可以任選圖表段落來檢視數據分布，也可以用長條圖或曲線圖來進行分析。

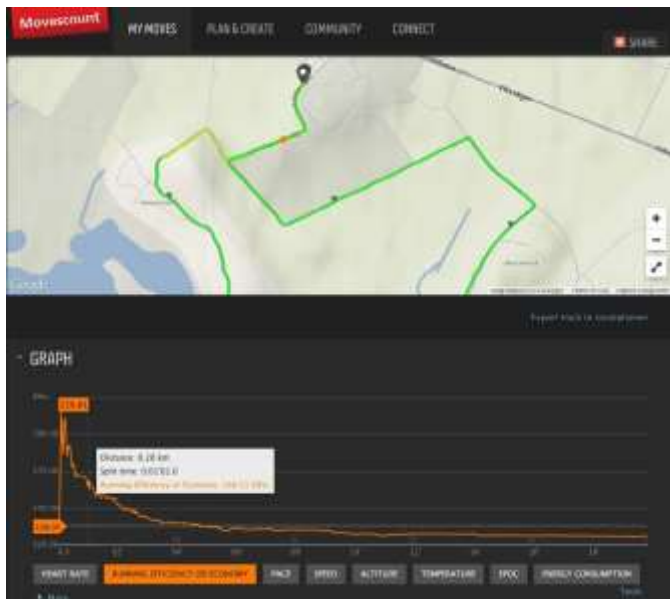


Figure 7 - Running Efficiency or Economy -App result shown in graph and on track.

圖 7 - 跑步的效率/經濟性 - App 結果追蹤與圖示

APP USER AGREEMENTS APP 使用者協議

When you create an App and store it in Movescount.com, you are obligated to accept the terms of service.

當您編寫了一個 App、並把它儲存到 Movescount.com 的時候，您便有接受 Suunto 服務條款的義務。

Likewise, when Movescount.com members save an App from App Zone, they are obligated to accept the end user agreement.

同樣地，當 Movescount.com 的會員從 App Zone 下載一款 App，他們也有接受使用者條款的義務。

[Suunto App Licence Agreement](http://www.movescount.com/suunto_app_license_agreement)

http://www.movescount.com/suunto_app_license_agreement

APP CODE REFERENCE APP 說明代碼參考

RESULT 值

The App value is placed in RESULT. RESULT is shown on device screen as a number or graph depending how the App user has customized the sport mode.

RESULT 是 APP 要呈現的值。RESULT 會呈現在裝置螢幕上可顯示為數字或圖表，端視使用者的自訂運動模式而定。

For example, the following line would show "100" on the screen:

舉例而言，下面的程式會在螢幕上顯示"100"：

```
RESULT = 100;
```

And the following would show current speed on the screen:
而下面的程式會顯示現在的速度：


```
RESULT = SUUNTO_SPEED;
```

RESULT can be shown in four different formats, available from the Format drop-down list. Depending on the option selected, the value 100 could be displayed as:

RESULT 的數值有四種顯示格式，可由 **Format** 的下拉式選單選擇；依照選擇的不同，“100” 會有下列不同的顯示格式：

time: 0:01'40

0 decimal: 100

1 decimal: 100.0

2 decimals: 100.00

PREFIX AND POSTFIX 前綴字與後綴字

The App results are shown with a prefix and postfix texts. These can be modified in App Designer modify element. It is also possible to modify the prefix and postfix in advanced mode. The syntax for prefix and postfix is as follows:

App 數據結果顯示可以加上前綴字與/或後綴字。前綴字與後綴字可以在 App 設計平台的參數修改處或進階模式中作修改。前綴字與後綴字的語法如下：

```
prefix="A";  
postfix="B";
```

The device screen would display "A [RESULT] B".

裝置螢幕將會顯示：“A [RESULT] B”。



Figure 8 - Result modification with format, prefix and postfix tools.

圖 8 – 結果(RESULT)格式設定、前綴字與後綴字工具。

CASE SENSITIVITY 大小寫有別

App Script is case sensitive.

App 語法必須注意字母大小寫是有別的。

For instance, the following script shows the result on the screen as "0."

舉例來說，下面這行程式結果會顯示為"0"：

```
RESULT = 0;
```

However, this script does NOT work:

但是，下面這行程式卻是無法運作的。

```
result = 0;
```

```
Error information: "Whoops! Please check the formula."
```

SEMICOLON 以分號結尾

Every statement needs to be terminated with a semicolon “;” The following script shows the result on the screen as “0.”

每一段敘述都需要以分號(;)結尾。下面這行程式結果會顯示為“0”：

```
RESULT=0;
```

However, this script does NOT work:

但是，下面這行程式卻是無法運作的。

```
RESULT=0
```

```
Error information: "Compilation has at least one error on line 2."
```

ERRORS 錯誤訊息

When creating a new App, you can verify the validity of the script at any stage. The verification will run the code and provide information about errors.

編寫 App 的時候，您可以隨時驗證語法的正確性。驗證程序在運行 App 之後，若發現語法錯誤，會提供相關的錯誤訊息。

Some of the errors that may be reported include:

常見的語法錯誤訊息包括：

“Compiled binary is too large for any supported device.”

Error indicates the App is using more memory than what is available in the device. Reduce the number of variables used in the script.

“編譯後二元碼太大，裝置無法容納。” 表示 App 所需要的記憶體空間大於裝置內的可用記憶體空間；需要減少語法中使用的變數數目。

“The app is used in device which does not support the chosen variables or functions.”

Error indicates the App has functions/variables that are not supported by the selected device.

“App 適用的裝置不支援語法中使用的變量或函數。” 表示 APP 所使用的函數/變量不被選用的裝置所支援。

。

“RESULT must be the output variable.”

Error indicates RESULT is not defined. Define RESULT in your syntax.

“RESULT 必須是輸出變數。” 表示 RESULT 未被定義；需要在語法中加以定義。

“Unsupported input variable used.”

Error indicates App result is not a number. This Error is given if RESULT is a text, or the text is further used as a variable for a function.

“使用了不支援的輸入變數。” 表示 App 運算結果不是數字。可能 RESULT 是文字，或可能被用於函數中變數的文字。

“Unsupported function used.”

Error indicates the App has a function that is not supported. This error is given if the function name is misspelled, for example.

“使用了不支援的函數。” 表示 App 使用了一個不被支援的函數；也有可能只是函數的名稱拼錯了之類的。

“Compilation has at least one error.”

Error indicates that during App compilation there was at least one error. Check the code.

“語法編譯至少有一個錯誤。” 表示 App 的語法編譯至少有一個錯誤，需要重新檢視語法。

“Error in buffered variable on line 2”

Error indicates the variable buffer size is too big. For example `SUUNTO_BIKE_POWER_AVG[900];` will create an error because the size of the buffer is larger than 30 seconds.

“第二行變數緩衝區錯誤。” 表示變數的緩衝區過大。例如：`SUUNTO_BIKE_POWER_AVG[900];` 的運算結果會顯示錯誤，因為緩衝區大於 30 秒。

COMMENTS 註解

Comments must use the following syntax:

註解必須使用下列語法：

```
/* here is a comment (註解寫在這裡)*/
```

Comments are shown in light gray text.

註解以淺灰色文字顯示。

MATH FUNCTIONS 數學函數

Math functions offer various calculation methods to achieve the App result.

數學函數提供多樣的計算方法，以利達成所需的 App 運算結果。

TRIGONOMETRIC FUNCTIONS 三角函數

1. SIN(X)

Sin is equal to the length of a triangle's side opposite the angle divided by length of the hypotenuse.

Sin(x) 等於 x 角的對邊長度除以斜邊長度。

Script(語法) : retVal = Suunto.sin(argIn)

- argIn: type float, unit radians
型態為 float, 單位為弧度
- retVal: type float, no unit, range [-1,1]

型態為 float, 無單位, 數值區間為[-1,1]

2. COS(X)

Cos is equal to the length of the triangle side adjacent to angle divided by the length of the hypotenuse

Cos(x) 等於 x 角的等於鄰邊長度除以斜邊長度。

Script(語法) : retVal = Suunto.cos(argIn)

語法 : retVal = Suunto. cos (argIn)

- argIn: type float, unit radians
型態為 float, 單位為弧度
- retVal: type float, no unit, range [-1,1]
型態為 float, 無單位, 數值區間為[-1,1]

3. ATAN2(Y,X)

Atan2 returns the principal value of the arc tangent of y/x , expressed in radians.

Atan2 函數傳回的是數值 y/x 的反正切函數主值，單位為弧度。

Script(語法) : retVal = Suunto.atan2(argY,argX)

語法 : retVal = Suunto.atan2(argY,argX)

- argY: type float, no unit
 型態為 float, 無單位
- argX: type float, no unit
 型態為 float, 無單位
- retVal: type float, unit radians, range (-PI,PI]
 型態為 float, 單位為弧度, 數值區間為(-PI,PI]

4. TAN(X)

Tan returns the tangent of an angle of x radians

Tan(x) 函數傳回的是 x 角的正切函數值， x 單位為弧度。

Script(語法) : retVal = Suunto.tan(argIn)

- argIn: type float, unit radians
型態為 float, 單位為弧度
- retVal: type float, no unit, range [-INF,INF]
型態為 float, 無單位, 數值區間為[-INF, INF]

5. SIND(X)

Sind(X) returns the sine of the element in X, expressed in degrees.

Sind(X) 函數傳回的是 X 的正弦函數值，X 單位為度。

Script(語法) : retVal = Suunto.sind(argIn)

- argIn: type float, unit degrees
型態為 float, 單位為度
- retVal: type float, no unit, range [-1,1]
型態為 float, 無單位, 數值區間為[-1,1]

6. **COSD(X)**

Cosd(X) returns the cosine of the element of X, expressed in degrees.

Cosd(X) 函數傳回的是 X 的餘弦函數值，X 單位為度。

Script(語法) : retVal = Suunto.cosd(argIn)

- argIn: type float, unit degrees
型態為 float, 單位為度
- retVal: type float, no unit, range [-1,1]
型態為 float, 無單位, 數值區間為[-1,1]

7. **TAND(X)**

Tand(X) returns the tangent of the element of X, expressed in degrees.

Tand(X) 等於 X 的正切函數值，X 單位為度。

Script(語法) : retVal = Suunto.tand(argIn)

- argIn: type float, unit degrees
型態為 float, 單位為度
- retVal: type float, no unit, range [-INF,INF]
型態為 float, 無單位, 數值區間為[-INF, INF]

8. ATAND2(Y,X)

Atand2 returns the principal value of the arc tangent of y/x , expressed in degrees.

Atand2 函數傳回的是數值 y/x 的反正切函數主值，單位為度。

Script(語法) : retVal = Suunto.atand2(argY,argX)

- argY: type float, no unit
型態為 float, 無單位
- argX: type float, no unit
型態為 float, 無單位
- retVal: type float, unit degrees, range (-180,180]

型態為 float, 單位為弧度, 數值區間為(-180,180]

OTHER MATHEMATICAL FUNCTIONS 其他數學函數

1. Sqrt(X)

Sqrt returns the square root of x.

Sqrt 函數傳回的是 x 的平方根值。

Script(語法) : retVal = Suunto.sqrt(argX)

- argX: type float, no unit, range [0,INF]
型態為 float, 無單位, 數值區間為[0,INF]
- retVal: type float, no unit, range [0,INF]
型態為 float, 無單位, 數值區間為[0,INF]

2. EXP(X)

Exp returns the exponential value of x.

Exp 函數傳回的是 x 的指數值。

Script(語法) : retVal = Suunto.exp(argX)

- argX: type float, no unit, range [-INF,INF]
型態為 float, 無單位, 數值區間為[-INF,INF]
- retVal: type float, no unit, range [0,INF]
型態為 float, 無單位, 數值區間為[0,INF]

3. LOG(X)

Log returns the natural logarithm of x.

Log 函數傳回的是 x 的自然對數值。

Script(語法) : retVal = Suunto.log(argX)

- argX: type float, no unit, range [0,INF]
型態為 float, 無單位, 數值區間為[0,INF]
- retVal: type float, no unit, range [-INF,INF]

型態為 float, 無單位, 數值區間為[-INF,INF]

4. LOG10(X)

Log10 returns the common base 10 logarithm of x.

Log10 函數傳回的是 x 以 10 為底的對數值。

Script(語法) : retVal = Suunto.log10(argX)

- argX: type float, no unit, range [0,INF]
型態為 float, 無單位, 數值區間為[0,INF]
- retVal: type float, no unit, range [-INF,INF]
型態為 float, 無單位, 數值區間為[-INF,INF]

5. POW(X,Y)

Pow returns x to the power y. If y is not an integer, x must ≥ 0 .

Pow 函數傳回的是 x 的 y 次方。若 y 不為整數， x 必須大於等於 0。

Script(語法)：retVal = Suunto.pow(argX,argY)

- argX: type float, no unit, range [-INF,INF]
型態為 float, 無單位, 數值區間為[-INF,INF]
- argY: type float, no unit, range [-INF,INF]
型態為 float, 無單位, 數值區間為[-INF,INF]
- retVal: type float, no unit, range [-INF,INF]
型態為 float, 無單位, 數值區間為[-INF,INF]

6. HYPOT(X,Y)

Hypot returns the $\sqrt{x^2+y^2}$.

Hypot(X,Y)函數傳回的是 x 和 y 的合方根值。

Script(語法)：retVal = Suunto.hypot(argX,argY)

- argX: type float, no unit, range [-INF,INF]

型態為 float, 無單位, 數值區間為[-INF,INF]

- argY: type float, no unit, range [-INF,INF]
型態為 float, 無單位, 數值區間為[-INF,INF]
- retVal: type float, no unit, range [0,INF]
型態為 float, 無單位, 數值區間為[0,INF]

7. RAND()

Rand returns a pseudorandom integer between zero and one.

Rand 函數傳回的是 0 或 1 之間的隨機整數。

Jerry 注解, 可以想成丟硬幣的隨機選正反面

Script(語法) : retVal = Suunto.rand()

- retVal: type float, 'unit' integers, range [0,1]
型態為 float, '單位'整數, 數值區間為[0,1]

8. MOD(X,Y)

Mod returns $x - (y * \text{int}(x/y))$.

Mod 函數傳回的是 $x - (y * \text{int}(x/y))$ 的餘數值。

Script(語法) : retVal = Suunto.mod(argX,argY)

- argX: type float, no unit
 型態為 float, 無單位
- argY: type float, no unit
 型態為 float, 無單位
- retVal: type float, no unit, range (-argY,argY)
 型態為 float, 無單位, 數值區間為(-argY,argY)

9. ABS(X)

Abs returns $|x|$.

Abs 函數傳回的是 $|x|$ 值。

Script(語法) : retVal = Suunto.abs(argX)

- argX: type float, no unit, range [-INF,INF]
數型態為 float, 無單位,數值區間為[-INF,INF]
- retVal: type float, no unit, range [0,INF]
數型態為 float, 無單位,數值區間為[0,INF]

10. ROUND(X)

Returns the integer value that is nearest to X, with halfway cases rounded away from zero.

Round 函數傳回的是最接近 X 的整數值

Jerry 注釋, 四捨六入,五遠離零. 意思是, 0.5→1, 1.5→2, -0.5→-1, -1.5→-2

Script(語法) : retVal = Suunto.round(argX)

- argX: type float, no unit, range [-INF,INF]
型態為 float, 無單位,數值區間為[-INF,INF]
- retVal: type float, no unit, range [-INF,INF]

型態為 float, 無單位, 數值區間為[-INF,INF]

11. CEIL(X)

The function rounds X upward, returning the smallest integer value that is not less than X.

Ceil 函數由 x 值往上找、傳回不小於 x 的最小整數。

Script(語法) : retVal = Suunto.ceil(argX)

- argX: type float, no unit, range [-INF,INF]
型態為 float, 無單位, 數值區間為[-INF,INF]
- retVal: type float, no unit, range [-INF,INF]
型態為 float, 無單位, 數值區間為[-INF,INF]

12. FLOOR(X)

The function rounds X downward, returning the largest integer value that is not greater than X.

Floor 函數由 x 值往下找、傳回不大於 x 的最大整數。

Script(語法) : retVal = Suunto.floor(argX)

- argX: type float, no unit, range [-INF,INF]
型態為 float, 無單位, 數值區間為[-INF,INF]
- retVal: type float, no unit, range [-INF,INF]
型態為 float, 無單位, 數值區間為[-INF,INF]

SUUNTO FUNCTIONS SUUNTO 函數

1. ALARMBEEP()

The following syntax plays an alarm.

下列語法能使 Suunto 裝置發出警示聲。

```
Suunto.alarmBeep()
```

2. LIGHT()

The following syntax turns the backlight on in the device.

下列語法能開啟 Suunto 裝置的背光功能。

```
Suunto.light()
```

3. DISTANCE(LAT, LON)

Returns distance to the current location.

計算出起點至現在位置的距離。

Script(語法) : retVal = Suunto.distance(lat, lon)

- lat: type float, degrees, range [-90,90]
 型態為 float, 單位為度,數值區間為[-90,90]
- lon: type float, degrees, range [-180,180]
 型態為 float, 單位為度,數值區間為[-180,180]
- retVal: type float, meters, range 0,21000000]
 型態為 float, 單位為公尺,數值區間為[0,21000000]

4. **DISTANCE(LAT1, LON1, LAT2, LON2)**

Returns the distance between two locations.

計算出兩個位置點之間的距離。

Script(語法) : retVal = Suunto.distance(lat1, lon1, lat2, lon2)

- lat1: type float, degrees, range [-90,90]

型態為 float,單位為度,數值區間為[-90,90]

- lon1: type float, degrees, range [-180,180]
型態為 float, 單位為度,數值區間為[-180,180]
- lat2: type float, degrees, range [-90,90]
型態為 float, 單位為度,數值區間為[-90,90]
- lon2: type float, degrees, range [-180,180]
浮點數型態為 float, 單位為度,數值區間為[-180,180]
- retVal: type float, meters, range 0,21000000]
型態為 float, 單位為公尺,數值區間為[0,21000000]

5. HEADING(LAT, LON)

Returns the heading from current location to given location.

顯示從目前位置到指定位置的方向。

Script(語法) : retVal = Suunto.heading(lat, lon)

- lat: type float, degrees, range [-90,90]
型態為 float, 單位為度,數值區間為[-90,90]

- lon: type float, degrees, range [-180,180]
型態為 float, 單位為度,數值區間為[-180,180]
- retVal: type float, degrees, range [0,360]
型態為 float, 單位為度,數值區間為[0,360]

6. HEADING(LAT1, LON1, LAT2, LON2)

Returns the heading from current location to given location.

顯示從指定位置 1 到指定位置 2 的方向。

Script(語法) : retVal = Suunto.heading(lat1, lon1, lat2, lon2)

- lat1: type float, degrees, range [-90,90]
型態為 float, 單位為度,數值區間為[-90,90]
- lon1: type float, degrees, range [-180,180]
型態為 float, 單位為度,數值區間為[-180,180]
- lat2: type float, degrees, range [-90,90]
型態為 float, 單位為度,數值區間為[-90,90]

- lon2: type float, degrees, range [-180,180]
型態為 float, 單位為度, 數值區間為[-180,180]
- retVal: type float, degrees, range [0,360]
型態為 float, 單位為度, 數值區間為[0,360]

OPERATORS AND STRUCTUR 運算子與語法結構

The available operators for an App are:

App 中可使用的運算子包括：

+	Addition	加
-	Subtraction	減
*	Multiplication	乘
/	Division	除

TIP! When creating mathematical functions in graphic mode, you can switch to the code editor in advanced mode. This will bring the already created function into the code editor automatically.

小訣竅！在圖形模式中需要建立數學運算式的時候，你可以先切換到進階模式的程式編輯器，因為程式編輯器會將內建的函數自動帶入。

RELATIONAL OPERATORS 關係運算子

The following comparisons are available:

以下是可供使用的比較關係運算子：

<code>==</code>	<i>Equal</i>	等於
<code>!=</code>	<i>Not equal</i>	不等於
<code>></code>	<i>Greater than</i>	大於
<code>>=</code>	<i>Greater than or equal to</i>	大於等於
<code><</code>	<i>Less than</i>	小於
<code><=</code>	<i>Less than or equal to</i>	小於等於

LOGICAL OPERATORS 邏輯運算子

App Designer provides logical operators to join different conditions:

App 設計平台提供邏輯運算子以結合不同的條件：

	OR binary disjunction	或 (二元法)
&&	AND conjunction	且

Following code is an example how to use logical operators. In this example, an App has different states that are handled with various relational and logical operators.

下面是使用邏輯運算子的範例。範例中，App 因為不同關係運算子與邏輯運算子的計算，產生了不同的狀態。



Figure 9 - Example of relational and logical operators in use

圖 9 – 關係運算子與邏輯運算子的使用範例

LOGICAL STRUCTURES 邏輯結構

With the advanced mode editor, you can create logical structures such as:

在進階模式下，你可以編寫如下的邏輯結構：

```
if (is  
true){ Do  
this  
}
```

The editor also supports the if...else statements

進階編輯器也支援 "if...else" 敘述

```
if (is  
true){ Do  
this;  
}  
else if (is  
true){ Then do  
this;  
}
```


In the following example, the App shows different heart rate target levels based on the distance. The first case shows several if statements where the last true statement is shown on the device screen. The target heart rate after the first 3km is 160, so the screen shows 43 beats below the target. The next screen shows the same logic, but with if...else statements. When the first if else statement is true, then the remaining code is not processed and the target heart rate level is set to 130.

下面的範例中，App 依據距離顯示不同的目標心率。第一個例子顯示了幾個 if 敘述，最後一個為真的陳述會顯示在屏幕上。開頭 3 公里的目標心率是 160，螢幕顯示心率为 43，低於目標值。第二個螢幕顯示相同的邏輯，但是使用的是“if...else”敘述。當第一個“if...else”敘述為真時，程式的其餘部分便略過不運行，最終目標心率顯示為 130。



Figure 10 - Several if statements are true and the last if result is shown. Several else if statements are true results first condition to be shown.

圖 10 – 使用數個 “if” 敘述，顯示最後一個結果；有數個 “else if” 敘述為真，顯示第一個條件。

VARIABLES 變數

OWN VARIABLES 自有變數

Variables are declared separately. Variables can only store numbers. Each variable that is created needs to have a default value. Variables can be used within the code editor in advanced mode as well as in graphic mode. Variables can be used, for example, as counters when building the App logic.

變數的定義需要單獨宣告。變數僅能儲存數字。創建的每個變數都需要指定一個預設值。變數可以在進階模式的程式編輯器中使用，也可以在圖形模式中使用。舉例來說，變數可以在建立 App 邏輯的時候作為計數器使用。



Figure 11 - Own variables used in App Designer and in code editor

圖 11 – 自有變數用在 App 編輯器中與程式編輯器中。

DIFFERENT WATCH VARIABLE TYPES 各類錶用變數

There are over 200 variables available for designing Apps that follow these basic principles:

設計 App 時有超過 200 個變數可供選用，它們都遵循下列的規則：

REAL TIME VALUES 即時值

These can be current, cumulative or average values, including current average (e.g. average speed at the moment).

這些變數包括現值、累計值及平均值，也包含即時平均值（例如即時的平均速度）。

LAP VALUES 圈值

Such as average temperature during the lap. Laps may be created manually or automatically (e.g. 1km autolap). Lap values refer to current lap.

如：圈中平均溫度。圈數可以手動或自動產生（例如：1 公里自動計圈）。圈值是指當前的圈。

PREVIOUS LAP VALUE 前一圈值

Such as average HR from previous lap. These lap values are taken from the lap that ended when current lap started.
如：前一圈的平均心率。這些數值取自當前的圈開始前的那一圈。

INSTANT VALUES 瞬時值

Such as instant altitude. Instant value is the value that was recorded when the lap ended. With instant altitude, that would be the altitude when lap button was last pressed.
如：瞬時高度。瞬時值是在一圈結束時紀錄的值。以瞬時高度為例，它就是最後一次按下 lap 按鈕時的高度值。

TAIL VALUE TAIL 值

Such as average_speed [30]

如：average_speed [30]。

These are values during the time period "Tail." This time period is by default 30 seconds, but the duration can be changed shortened. Tail values can be split into five categories:

Tail 應該是前一區間。例如，若 tail 是 30 秒，每分鐘的第 0 秒到第 29 秒的 Tail 指的是前一分鐘的 30 秒到 59 秒這個區間，每分鐘的 30~59 秒，Tail 指的是這一分鐘的前 0~29 秒這個區間。這些是在 "Tail" 時間區段內獲得的值。Tail 時間預設為 30 秒，但是可以自行設定縮短。Tail 值可分為五大類：

- Average: average value during the tail
均值：Tail 時間內的平均值
- Max: maximum value during the tail
極大值：Tail 時間的極大值
- Min: lowest or smallest value during the tail
極小值：Tail 時間內的最低值或最小值

- Diff: difference between first and last value, that is, the value at the first second of the tail compared to the value at the last second of the tail

差值：首值與末值之差。首值是 Tail 時間的第一秒所記錄的值；末值是 Tail 時間的最後一秒所記錄的值。

- Total: total value accumulated during the tail

總值：Tail 時間內累計的總值。

NOTE! Using different tail lengths within one App can lead to a very large App. So, depending to the device, such a large App might not fit into the device memory.

備註：在同一個 App 中若使用不同長度的 Tail 時間，會使得 App 變得很大，甚至大到裝不進 Suunto 裝置中 – 依裝置的記憶體大小而定。

Type of the variables can be:

變數的種類包括：

$s = \text{seconds}$

秒

km/h = kilometers/hour

公里/小時

min/km = minutes per kilometer

分鐘/公里

km = kilometers

公里

m = meters

公尺

num = number 0,1,2->

數字

bpm = beats per minute (heart rate)

每分鐘次數 (心跳)

PTE = peak training effect

峰值培訓效果

(1-5, with one decimal i.e. 2.8)

hPa = pressure

百帕 (大氣壓力)

C = Celsius (temperature)

攝氏 (溫度)

W = Watts (power)

瓦特 (功率)

rpm = rounds per minute

每分鐘圈數

kcal = kilocalories (energy)

千卡 (能量)

kg = weight

公斤 (重量)

h = hours

小時

WATCH VARIABLES 錶用變數

Variable

NAME_IN_EDITOR

Min - Max values in watch

(Values used in App designer simulation in parentheses.)

SPEED

Speed <i>SUUNTO_SPEED</i>	0	277 <i>(100)</i>	(km/h)
Average speed <i>SUUNTO_AVG_SPD</i>	0	277 <i>(100)</i>	(km/h)
Maximum speed <i>SUUNTO_MAX_SPD</i>	0	277 <i>(100)</i>	(km/h)
Pace <i>SUUNTO_PACE</i>	30 <i>(17)</i>	0.05 <i>(0.6)</i>	(min/km)
Lap avg speed <i>SUUNTO_LAP_SPD</i>	0	277 <i>(100)</i>	(km/h)
Lap maximum speed <i>SUUNTO_LAP_MAX_SPD</i>	0	277 <i>(100)</i>	(km/h)

Lap avg pace <i>SUUNTO_LAP_PACE</i>	30 (17)	0.05 (0.6)	(min/km)
Previous lap avg speed <i>SUUNTO_LAP_SPD_PREV</i>	0	277 (100)	(km/h)
Previous lap max speed <i>SUUNTO_LAP_MAX_SPD_PREV</i>	0	277 (100)	(km/h)
Previous lap avg pace <i>SUUNTO_LAP_PACE_PREV</i>	30 (17)	0.05 (0.6)	(min/km)
Instant speed <i>SUUNTO_SPEED_PREV</i>	0	277 (100)	(km/h)
Instant avg speed <i>SUUNTO_AVG_SPD_PREV</i>	0	277 (100)	(km/h)
Instant pace <i>SUUNTO_PACE_PREV</i>	30 (17)	0.05 (0.6)	(min/km)
Tail avg speed[30sec] <i>SUUNTO_SPEED_AVG</i>	0	277 (100)	(km/h)
Tail max speed[30sec] <i>SUUNTO_SPEED_MAX</i>	0	277 (100)	(km/h)
Tail min speed[30sec] <i>SUUNTO_SPEED_MIN</i>	0	277 (100)	(km/h)
Tail diff speed[30sec] <i>SUUNTO_SPEED_DIFF</i>	0	277 (100)	(km/h)

Tail avg pace[30sec] <i>SUUNTO_PACE_AVG</i>	30 (17)	0.05 (0.6)	(min/km)
Tail max pace[30sec] <i>SUUNTO_PACE_MAX</i>	30 (17)	0.05 (0.6)	(min/km)
Tail min pace[30sec] <i>SUUNTO_PACE_MIN</i>	30 (17)	0.05 (0.6)	(min/km)
Tail diff pace[30sec] <i>SUUNTO_PACE_DIFF</i>	30 (17)	0.05 (0.6)	(min/km)

DISTANCE/GPS

Distance <i>SUUNTO_DISTANCE</i>	0	9999 (50)	(km)
Lap distance <i>SUUNTO_LAP_DISTANCE</i>	0	9999 (50)	(km)
Autolap distance <i>SUUNTO_AUTOLAP_DISTANCE</i>	0	9999 (50)	(km)
Manual lap distance <i>SUUNTO_MANUAL_LAP_DISTANCE</i>	0	9999 (50)	(km)
Previous lap distance <i>SUUNTO_LAP_DISTANCE_PREV</i>	0	9999 (50)	(km)

Instant distance <i>SUUNTO_DISTANCE_PREV</i>	0	9999 (50)	(km)
Tail total distance[30sec] <i>SUUNTO_DISTANCE_TOT</i>	0	9999 (50)	(km)
Latitude <i>SUUNTO_GPS_LATITUDE</i>	-90	90	(degrees)
Longitude <i>SUUNTO_GPS_LONGITUDE</i>	-180	180	(degrees)
GPS state <i>SUUNTO_GPS_STATE</i>	0	100	(0=off, 1=starting, 2= hibernating, 3=loading SGEE, 4=activating 5-100= active)
GPS altitude <i>SUUNTO_GPS_ALTITUDE</i>	-10000 (0)	20000 (5000)	(m)
GPS heading <i>SUUNTO_GPS_HEADING</i>	0	360	(Degrees)

Navigation mode <i>SUUNTO_NAVIGATION_MODE</i>	0	2	(0=off, 1=POI, 2= route)
Navigation distance to next waypoint <i>SUUNTO_DISTANCE_TO_NEXT_TARGET</i>	0	41000 (50)	(km)
Navigation distance to last waypoint <i>SUUNTO_DISTANCE_TO_PREVIOUS_TARGET</i>	0	41000 (50)	(km)

HR

Heart rate <i>SUUNTO_HR</i>	30	240 (229)	(bpm)
Average heart rate <i>SUUNTO_AVG_HR</i>	30	240 (229)	(bpm)
Maximum heart rate <i>SUUNTO_MAX_HR</i>	30	240 (229)	(bpm)
Peak Training Effect <i>SUUNTO_PEAKE</i>	1	5	(PTE)

Energy consumption <i>SUUNTO_ENERGY</i>	0	60000 (3000)	(kcal)
Lap avg heart rate <i>SUUNTO_LAP_AVG_HR</i>	30	240 (229)	(bpm)
Lap max heart rate <i>SUUNTO_LAP_MAX_HR</i>	30	240 (229)	(bpm)
Lap energy consumption <i>SUUNTO_LAP_ENERGY</i>	0	60000 (3000)	(kcal)
Previous lap avg heart rate <i>SUUNTO_LAP_AVG_HR_PREV</i>	30	240 (229)	(bpm)
Previous lap max heart rate <i>SUUNTO_LAP_MAX_HR_PREV</i>	30	240 (229)	(bpm)
Previous lap energy consumption <i>SUUNTO_LAP_ENERGY_PREV</i>	0	60000 (3000)	(kcal)
Instant heart rate <i>SUUNTO_HR_PREV</i>	30	240 (229)	(bpm)
Instant avg heart rate <i>SUUNTO_AVG_HR_PREV</i>	30	240 (229)	(bpm)
Instant Peak Training Effect <i>SUUNTO_PEAKE_PREV</i>	1	5	(PTE)
Instant energy consumption <i>SUUNTO_ENERGY_PREV</i>	0	60000 (3000)	(kcal)

Tail avg heart rate[30sec] <i>SUUNTO_HR_AVG</i>	30	240 (229)	(bpm)
Tail max heart rate[30sec] <i>SUUNTO_HR_MAX</i>	30	240 (229)	(bpm)
Tail min heart rate[30sec] <i>SUUNTO_HR_MIN</i>	30	240 (229)	(bpm)
Tail diff heart rate[30sec] <i>SUUNTO_HR_DIFF</i>	30	240 (229)	(bpm)

ALTITUDE

Altitude <i>SUUNTO_ALTI</i>	-500 (0)	9000 (5000)	(m)
Ascent <i>SUUNTO_ASCENT</i>	0	65535 (2000)	(m)
Descent <i>SUUNTO_DESCENT</i>	0	65535 (2000)	(m)
Ascent time <i>SUUNTO_ASCENT_TIME</i>	0	2147483647 (3600)	(s)

Descent time <i>SUUNTO_DESCENT_TIME</i>	0	2147483647 (3600)	(s)
Vertical speed <i>SUUNTO_VERTICAL_SPD</i>	-212400 (-100)	212400 (100)	(m/min)
Lap ascent <i>SUUNTO_LAP_ASCENT</i>	0	65535 (2000)	(m)
Lap descent <i>SUUNTO_LAP_DESCENT</i>	0	65535 (2000)	(m)
Lap ascent time <i>SUUNTO_LAP_ASCENT_TIME</i>	0	2147483647 (3600)	(s)
Lap descent time <i>SUUNTO_LAP_DESCENT_TIME</i>	0	2147483647 (3600)	(s)
Previous lap ascent <i>SUUNTO_LAP_ASCENT_PREV</i>	0	65535 (2000)	(m)
Previous lap descent <i>SUUNTO_LAP_DESCENT_PREV</i>	0	65535 (2000)	(m)
Previous lap ascent time <i>SUUNTO_LAP_ASCENT_TIME_PREV</i>	0	2147483647 (3600)	(s)
Previous lap descent time <i>SUUNTO_LAP_DESCENT_TIME_PREV</i>	0	2147483647 (3600)	(s)

Instant altitude <i>SUUNTO_ALTI_PREV</i>	-500 (0)	9000 (5000)	(m)
Instant ascent <i>SUUNTO_ASCENT_PREV</i>	0	65535 (2000)	(m)
Instant descent <i>SUUNTO_DESCENT_PREV</i>	0	65535 (2000)	(m)
Tail total ascent[30sec] <i>SUUNTO_ASCENT_TOT</i>	0	65535 (2000)	(m)
Tail total descent[30sec] <i>SUUNTO_DESCENT_TOT</i>	0	65535 (2000)	(m)
Tail total ascent time[30sec] <i>SUUNTO_ASCENT_TIME_TOT</i>	0	2147483647 (3600)	(s)
Tail total descent time[30sec] <i>SUUNTO_DESCENT_TIME_TOT</i>	0	2147483647 (3600)	(s)
Tail avg altitude[30sec] <i>SUUNTO_ALTI_AVG</i>	-500 (0)	9000 (5000)	(m)
Tail max altitude[30sec] <i>SUUNTO_ALTI_MAX</i>	-500 (0)	9000 (5000)	(m)
Tail min altitude[30sec] <i>SUUNTO_ALTI_MIN</i>	-500 (0)	9000 (5000)	(m)
Tail diff altitude[30sec] <i>SUUNTO_ALTI_DIFF</i>	-500 (0)	9000 (5000)	(m)

Tail avg vertical speed[30sec] <i>SUUNTO_VERTICAL_SPD_AVG</i>	-212400 (-100)	212400 (100)	(m/min)
Tail max vertical speed[30sec] <i>SUUNTO_VERTICAL_SPD_MAX</i>	-212400 (-100)	212400 (100)	(m/min)
Tail min vertical speed[30sec] <i>SUUNTO_VERTICAL_SPD_MIN</i>	-212400 (-100)	212400 (100)	(m/min)
Tail diff vertical speed[30sec] <i>SUUNTO_VERTICAL_SPD_DIFF</i>	-212400 (-100)	212400 (100)	(m/min)

ENVIRONMENT

Pressure <i>SUUNTO_PRESSURE</i>	950 (850)	1060 (1050)	(hPa - sealevel)
Temperature <i>SUUNTO_TEMP</i>	-20 (-30)	60 (40)	(C)
Minimum temperature <i>SUUNTO_MIN_TEMP</i>	-20 (-30)	60 (40)	(C)
Maximum temperature <i>SUUNTO_MAX_TEMP</i>	-20 (-30)	60 (40)	(C)
Instant pressure <i>SUUNTO_PRESSURE_PREV</i>	950 (850)	1060 (1050)	(hPa - sealevel)

Instant temperature <i>SUUNTO_TEMP_PREV</i>	-20 (-30)	60 (40)	(C)
Tail avg pressure[30sec] <i>SUUNTO_PRESSURE_AVG</i>	950 (850)	1060 (1050)	(hPa - sealevel)
Tail max pressure[30sec] <i>SUUNTO_PRESSURE_MAX</i>	950 (850)	1060 (1050)	(hPa - sealevel)
Tail min pressure[30sec] <i>SUUNTO_PRESSURE_MIN</i>	950 (850)	1060 (1050)	(hPa - sealevel)
Tail diff pressure[30sec] <i>SUUNTO_PRESSURE_DIFF</i>	950 (850)	1060 (1050)	(hPa - sealevel)
Tail avg temperature[30sec] <i>SUUNTO_TEMP_AVG</i>	-20 (-30)	60 (40)	(C)
Tail max temperature[30sec] <i>SUUNTO_TEMP_MAX</i>	-20 (-30)	60 (40)	(C)
Tail min temperature[30sec] <i>SUUNTO_TEMP_MIN</i>	-20 (-30)	60 (40)	(C)
Tail diff temperature[30sec] <i>SUUNTO_TEMP_DIFF</i>	-20 (-30)	60 (40)	(C)

TIME

Duration <i>SUUNTO_DURATION</i>	0	921600 (18000)	(seconds)
Time <i>SUUNTO_TIME</i>	0	86400	(sec from midnight)
Date <i>SUUNTO_DAYS_AFTER_1_1_2000</i>	4750	6000	(days since 1.1.2000)
Lap number <i>SUUNTO_LAP_NUMBER</i>	0	65535 (50)	(number)
Lap duration <i>SUUNTO_LAP_DURATION</i>	0	921600 (18000)	(seconds)
Autolap duration <i>SUUNTO_AUTOLAP_DURATION</i>	0	921600 (18000)	(seconds)
Manual lap duration <i>SUUNTO_MANUAL_LAP_DURATION</i>	0	921600 (18000)	(seconds)
Previous lap number <i>SUUNTO_LAP_NUMBER_PREV</i>	0	65535 (50)	(number)
Previous lap duration <i>SUUNTO_LAP_DURATION_PREV</i>	0	921600 (18000)	(seconds)

Instant time <i>SUUNTO_TIME_PREV</i>	0	86400	(sec from midnight)
Instant duration <i>SUUNTO_DURATION_PREV</i>	0	921600 (18000)	(seconds)
Fastest distance <i>SUUNTO_FASTEST_DISTANCE</i>	0	65535 (18000)	(s/km OR s/mi)
Fastest distance unit <i>SUUNTO_FASTEST_DISTANCE_UNIT</i>	0	1	(0=km, 1=mi)

SWIMMING

Pool length <i>SUUNTO_SWIMMING_POOL_LENGTH</i>	0 (25)	10000 (200)	(m)
Strokes <i>SUUNTO_SWIMMING_STROKES</i>	0	2147483647 (100)	(nbr of strokes)
Rest time <i>SUUNTO_SWIMMING_REST_TIME</i>	0	2147483647 (720)	(seconds)
Previous pool length duration <i>SUUNTO_SWIMMING_PREVIOUS_POOL_LENGTH_DURATION</i>	0	2147483647 (120)	(seconds)
Previous pool length strokes <i>SUUNTO_SWIMMING_PREVIOUS_POOL_LENGTH_STROKES</i>	0	2147483647 (100)	(nbr of strokes)

Interval duration <i>SUUNTO_SWIMMING_INTERVAL_DURATION</i>	0	2147483647 (3600) (seconds)
Interval distance <i>SUUNTO_SWIMMING_INTERVAL_DISTANCE</i>	0	2147483647 (1000) (m)
Interval strokes <i>SUUNTO_SWIMMING_INTERVAL_STROKES</i>	0	2147483647 (1000) (nbr of strokes)
Tail avg strokes[30sec] <i>SUUNTO_SWIMMING_STROKES_AVG</i>	0	2147483647 (1000) (nbr of strokes)
Tail max strokes[30sec] <i>SUUNTO_SWIMMING_STROKES_MAX</i>	0	2147483647 (1000) (nbr of strokes)
Tail min strokes[30sec] <i>SUUNTO_SWIMMING_STROKES_MIN</i>	0	2147483647 (1000) (nbr of strokes)
Tail diff strokes[30sec] <i>SUUNTO_SWIMMING_STROKES_DIFF</i>	0	2147483647 (1000) (nbr of strokes)
Previous pool length style <i>SUUNTO_SWIMMING_PREVIOUS_POOL_LENGTH_STYLE</i>	0	5 (0=other,1=butterfly, 2=backstroke, 3=breaststroke, 4=freestyle, 5=drill)

POWER

Bike power <i>SUUNTO_BIKE_POWER</i>	0	9999 (500)	(W)
Average bike power <i>SUUNTO_BIKE_AVG_POWER</i>	0	9999 (500)	(W)
Maximum bike power <i>SUUNTO_BIKE_MAX_POWER</i>	0	9999 (500)	(W)
Lap avg bike power <i>SUUNTO_LAP_BIKE_AVG_POWER</i>	0	9999 (500)	(W)
Lap max bike power <i>SUUNTO_LAP_BIKE_MAX_POWER</i>	0	9999 (500)	(W)
Tail avg power[30sec] <i>SUUNTO_BIKE_POWER_AVG</i>	0	9999 (500)	(W)
Tail max power[30sec] <i>SUUNTO_BIKE_POWER_MAX</i>	0	9999 (500)	(W)
Tail min power[30sec] <i>SUUNTO_BIKE_POWER_MIN</i>	0	9999 (500)	(W)
Tail diff power[30sec] <i>SUUNTO_BIKE_POWER_DIFF</i>	0	9999 (500)	(W)
Bike power connected <i>SUUNTO_BIKE_POWER_CONNECTED</i>	0	1	(0=not connected, 1=connecte d)

CADENCE

Cadence <i>SUUNTO_CADENCE</i>	0	240 (199)	(rpm)
Average cadence <i>SUUNTO_AVG_CADENCE</i>	0	240 (199)	(rpm)
Maximum cadence <i>SUUNTO_MAX_CADENCE</i>	0	240 (199)	(rpm)
Lap avg cadence <i>SUUNTO_LAP_AVG_CADENCE</i>	0	240 (199)	(rpm)
Lap max cadence <i>SUUNTO_LAP_AVG_CADENCE_PREV</i>	0	240 (199)	(rpm)
Previous lap avg cadence <i>SUUNTO_LAP_MAX_CADENCE</i>	0	240 (199)	(rpm)
Previous lap max cadence <i>SUUNTO_LAP_MAX_CADENCE_PREV</i>	0	240 (199)	(rpm)
Instant cadence <i>SUUNTO_CADENCE_PREV</i>	0	240 (199)	(rpm)
Instant avg cadence <i>SUUNTO_AVG_CADENCE_PREV</i>	0	240 (199)	(rpm)
Tail avg cadence[30sec] <i>SUUNTO_CADENCE_AVG</i>	0	240 (199)	(rpm)

Tail max cadence[30sec] <i>SUUNTO_CADENCE_MAX</i>	0	240 (199)	(rpm)
Tail min cadence[30sec] <i>SUUNTO_CADENCE_MIN</i>	0	240 (199)	(rpm)
Tail diff cadence[30sec] <i>SUUNTO_CADENCE_DIFF</i>	0	240 (199)	(rpm)

PERSONAL

User max heart rate <i>SUUNTO_USER_MAX_HR</i>	30	240 (229)	(bpm)
User rest heart rate <i>SUUNTO_USER_REST_HR</i>	30	240 (229)	(bpm)
User age <i>SUUNTO_USER_AGE</i>	0 (10)	150 (99)	(year)
User weight <i>SUUNTO_USER_WEIGHT</i>	30	200	(kg)
User activity class <i>SUUNTO_USER_ACTIVITY_CLASS</i>	1	10	
User height <i>SUUNTO_USER_HEIGHT</i>	89 (90)	241 (240)	(cm)

User gender <i>SUUNTO_USER_GENDER</i>	0	1	(0=female, 1=male)
User recovery time <i>SUUNTO_USER_RECOVERY_TIME</i>	0	15360 (120)	(h)
Activity type <i>SUUNTO_ACTIVITY_TYPE</i>	1	65535 (82)	(ID, list of activities below)

Not specified sport = 1	Multisport = 2	Run = 3
Cycling = 4	MountainBiking = 5	Swimming = 6
Skating = 8	Aerobics = 9	YogaPilates = 10
Trekking = 11	Walking = 12	Sailing = 13
Kayaking = 14	Rowing = 15	Climbing = 16
Indoor cycling = 17	Circuit training = 18	Triathlon = 19
Alpine skiing = 20	Snowboarding = 21	Crosscountry skiing = 22
Weight training = 23	Basketball = 24	Soccer = 25
Ice Hockey = 26	Volleyball = 27	Football = 28
Softball = 29	Cheerleading = 30	Baseball = 31

Tennis = 33	Badminton = 34	Table tennis = 35
Racquet ball = 36	Squash = 37	Combat sport = 38
Boxing = 39	Floorball = 40	Scuba diving = 51
Free diving = 52	Adventure Racing = 61	Bowling = 62
Cricket = 63	Cross trainer = 64	Dancing = 65
Golf = 66	Gymnastics = 67	Handball = 68
Horseback riding = 69	Ice Skating = 70	Indoor Rowing = 71
Canoeing = 72	Motorsports = 73	Mountaineering = 74
Orienteering = 75	Rugby = 76	Ski Touring = 78
Stretching = 79	Telemark skiing = 80	Track and Field = 81
Trail Running = 82	Open water swimming = 83	Nordic walking = 84
Snow shoeing = 85	Windsurfing/Surfing = 86	Kettlebell = 87
Roller skiing = 88	Standup paddling (SUP) = 89	Cross fit = 90
Kitesurfing/Kiting = 91	Paragliding = 92	Treadmill = 93
Frisbee = 94	Indoor training = 95	

NOT SUPPORTED FEATURES 不支援的功能

Some common software engineering features are not supported. Here are few examples.

有些常見的軟體開發功能是不支援的，例如：

- Arrays/vectors
矩陣/向量
- Switch statement
Switch 條件判斷式
- Loops (Actually the App is always in a loop because it's ran once per second.)
迴圈 (實際上，App 永遠都在迴圈中，因為它每秒運算一次)
- Objects
物件
- mean(x), min(x),max(x), std(x)