Dash, Watch Out for Rover

Challenge: Drive Dash as close as you can to the dog without hitting him.

Supplies: Tape measure and masking tape to mark your starting point.

Team Name: The most clever name will be used for tie breaking.

Code: Your program can *only* use the following drive function.

Enter number of seconds to drive Dash forward.

<p style="color:red; text-align:center; font-size:2em;">dash.forward_time(5)</p>

Judging: When your group is ready, the judge will place the pooch in the road at a random distance from Dash. You may not move the mutt or Dash once the pet is placed. You may *only change the value of time* in the Dash drive statement. Competition distance between bowser and the Dash will be reported in cm. ***Don't hit the hound!***

Watch out for Rover Program


```
EDITOR: DWATCH
PROGRAM LINE 0001
#Dash, watch out for Rover
from ww_dash import *

dash.forward_time(5)

# Try different values for
# time in seconds to drive.

# Press [Run] softkey to drive.

Fns…  a A #  Tools  Run  Files
```

from ww_dash import *
is available as a Python Module Add-on.

The picture can't be displayed.

The picture can't be displayed.

```
EDITOR: DWATCH
Func Ctl Ops List Type I/O Modul
1:math…
2:random…
3:time…
4:ti_system…
5:ti_plotlib…
6:ti_hub…
7:ti_rover…

Esc  Help          Add-On
```

[Left Arrow] to Modul menu

Press Add-on

```
EDITOR: DWATCH
IMPORTS
Add-On Module Imports
1:from ti_draw import *
2:from ti_image import *
3:from ww_dash import *

Esc
```

Select
3:from ww_dash import *

```
EDITOR: DWATCH1
PROGRAM LINE 0002
from ww_dash import *
_

Fns… a A
EDITOR: DWATCH
Func Ctl Ops List Type I/O Modul
1:math…
2:random…
3:time…
4:ti_system…
5:ti_plotlib…
6:ti_hub…
7:ti_rover…
8:ww_dash…

Esc  Help          Add-On
```

[Left Arrow] to Modul menu
And select ww_dash…

dash.forward_time( )
is available from the Dash Drive menu.
Select item 9:forward_time( )

```
EDITOR: DWATCH
Drive Settings I/O Commands
1:forward(distance)        unit
2:backward(distance)       unit
3:left(angle)           degrees
4:right(angle)          degrees
5:stay(time)            seconds
6:to_xy(x,y)
7:to_polar(r,theta)     degrees
8:to_angle(angle)       degrees
9:forward_time(time)    seconds
0↓backward_time(time)   seconds
Esc  Modul
```