



Unit 4: ドライブ機能

Skill Builder 1: Roverを動かそう!

このレッスンでは、TI-Innovator Roverロボット車両の操作を開始します。Roverを動かしたり、回転させたりします。

目標

- Roverを動かす
- Roverドライブをパターンで作成

PythonのTI-Innovator **Rover**コマンドは、すべて**rv**.で始まります。この構文は、**ti_rover**モジュールをプログラムにインポートする方法の結果です (**import ti_rover as rv**)。同じことが**ti_plotlib**モジュールにも当てはまります。この手法は、モジュール名のエイリアシング(aliasing)と呼ばれます(短い名前に置き換えます)。

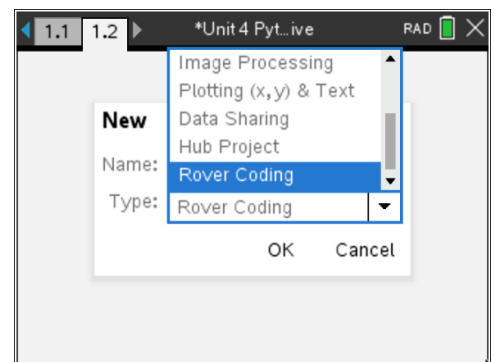
Roverを操作するプログラムを作成するときは、使いやすいコマンドを使って車両を制御しますが、他のTI-Innovator™ Hubコマンドと同様、舞台裏で多くのことが行われています。

TI-Nspire CX IIがRoverに接続されていること、そしてRoverの電源がオンになっていることを確認してください。



Teacher Tip: Pythonは初めてですか？ オンラインでエイリアシングモジュール Python(aliasing module Python)をチェックしてください。

1. 新規のPythonプログラムを開始し、プログラム名を入力した後、**Type**はドロップダウンメニューから**Rover Coding**テンプレートを選択します。

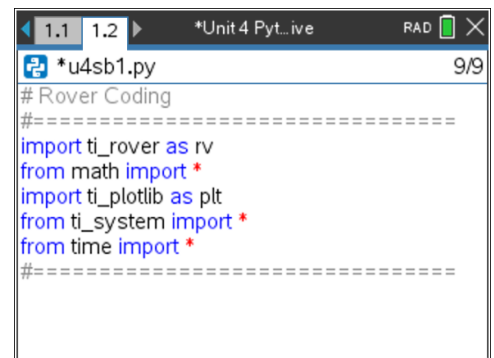


2. テンプレートには多くの**import**ステートメントがあります。便利な関数に簡単にアクセスできるようにするためです。

現在最も重要なのは、最初のステートメントです。

import ti_rover as rv

この形式のインポートステートメントは、**rv**を使う必要があることを示しています。**menu**からRover関数を選択すると、**rv**.は自動的に含まれます。



Teacher Tip: Pythonエディタにステートメント文字を入力することもできますが、Roverを制御する場合、**rv**.は必要であり、エラーの原因となる可能性があります。



3. Roverを前進させるには, **menu > TI Rover > Drive**(メニュー>TI Rover>ドライブ)により次のステートメントを得ます。

rv.forward(distance)

電卓メニューには**forward**の前の**rv.**がありません。プログラムには**rv.**が貼り付けられます。注意してください。これは、電卓メニューのスペースを節約するためです。

distance(距離)プロンプトはグリッド単位で測定され、それが何を意味するかはすぐに判断できます。**distance**として数字1を使ってプログラムを実行します。Roverが動いたら、大成功です!

しかし...Roverはどこまで移動しましたか。

```
*Unit4 Pyt...ive
RAD
*u4sb1.py 9/9
# Rover Coding
#-----
import ti_rover as rv
from math import *
import ti_plotlib as plt
from ti_system import *
from time import *
#-----
rv.forward(distance)
```

Teacher Tip: Roverの既定値の単位は10cm=1dm(デシメートル, 1mの1/10)=0.1mです。距離の単位はmで、角度の単位は度(°)です。**distance**プロンプトを削除すると、Roverは1m移動します。

4. 基本的なドライブコマンド**forward**(前進), **backward**(後退), **left**(左), **right**(右)をやってみましょう(右図参照)。

- 1 forward(distance)
- 2 backward(distance)
- 3 left(angle_degrees)
- 4 right(angle_degrees)

Teacher Tip: これは探究活動に適したところです。

5. 正方形の経路を運転するには :
 入力したドライブコマンドを消去します(またはコメント(#)にします)。
forループを使って、Roverドライブを正方形のパターンにします。
 次から始めます。

for index in range(size):

適当な**index**と**size**(サイズ)を使って正方形を作成し、ループblockに2つのステートメントのみを追加します。自分で試して、プログラムを実行してください。

```
*Unit4 Pyt...ive
RAD
*u4sb1.py 9/11
# Rover Coding
#-----
import ti_rover as rv
from math import *
import ti_plotlib as plt
from ti_system import *
from time import *
#-----
for index in range(size):
  ++block
```

6. 右図のようなものを書きましたか?

おめでとう! 作業内容を保存することを忘れないでください。
 他にどのようなパターンを作ることができますか。

```
*Unit4 Pyt...ive
RAD
*u4sb1.py 12/13
#-----
import ti_rover as rv
from math import *
import ti_plotlib as plt
from ti_system import *
from time import *
#-----
for i in range(4):
  ++rv.forward(1)
  ++rv.left(90)
  ++
```



Teacher Tip: Drive with Optionsメニュー(menu > TI Rover > Drive > Drive with Options(メニュー>TI Rover>ドライブ>オプション付きドライブ))には、多くのカスタマイズされたドライブステートメントがあります。距離、時間、速度に関連したもので、メートル、グリッド単位、ホイール回転(前方と後方)、度、ラジアン、勾配(角度)などの単位を使っています。これらのオプションはこのコースではカバーされていませんが、興味深い探究活動のチャンスです。