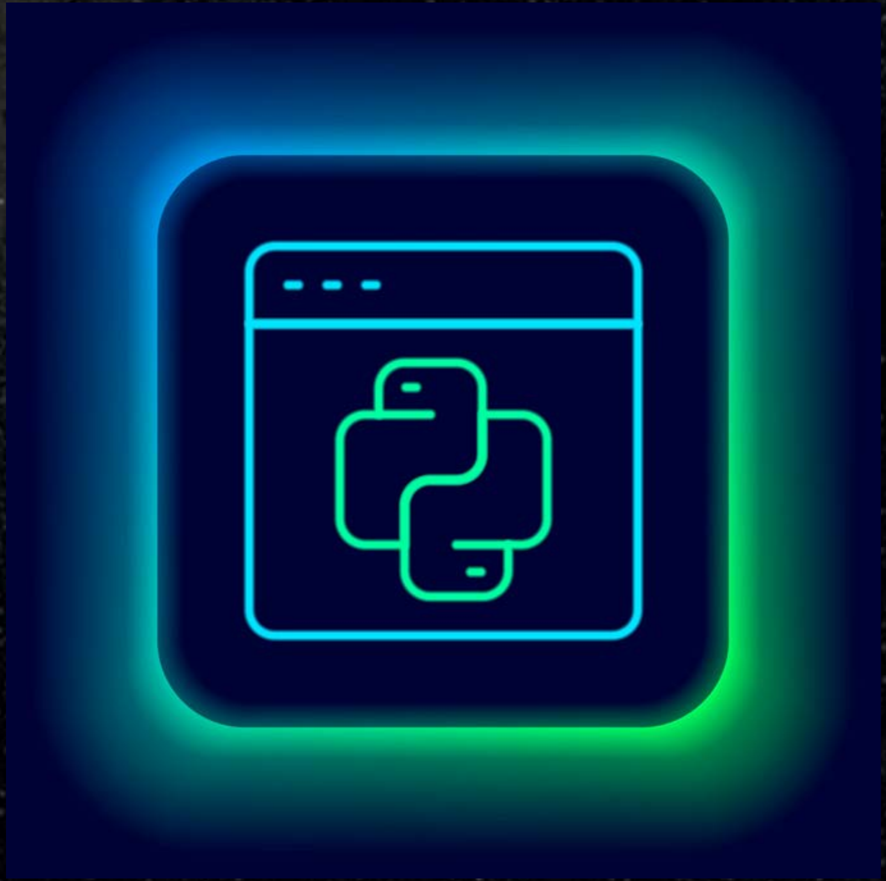
PYTHON

VARIABLES INTRODUCTION

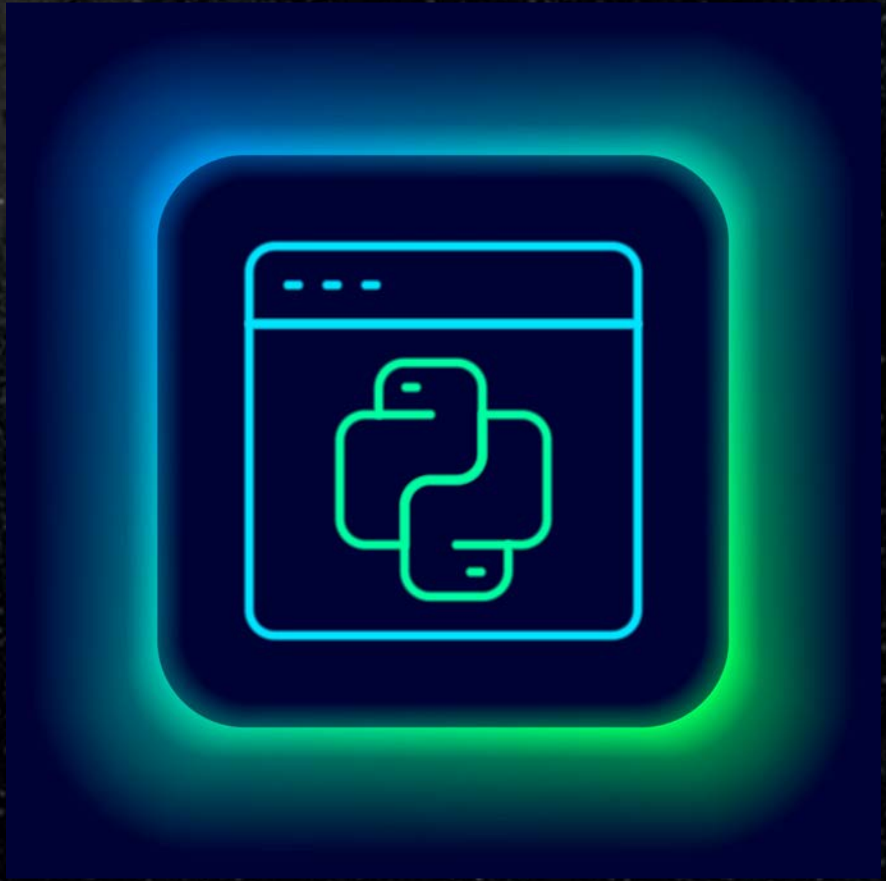
A RHYSING FOUNDATION PROGRAM SPONSORED BY UNITED AIRCRAFT TECHNOLOGIES

WHAT YOU WILL LEARN



WHAT YOU WILL LEARN

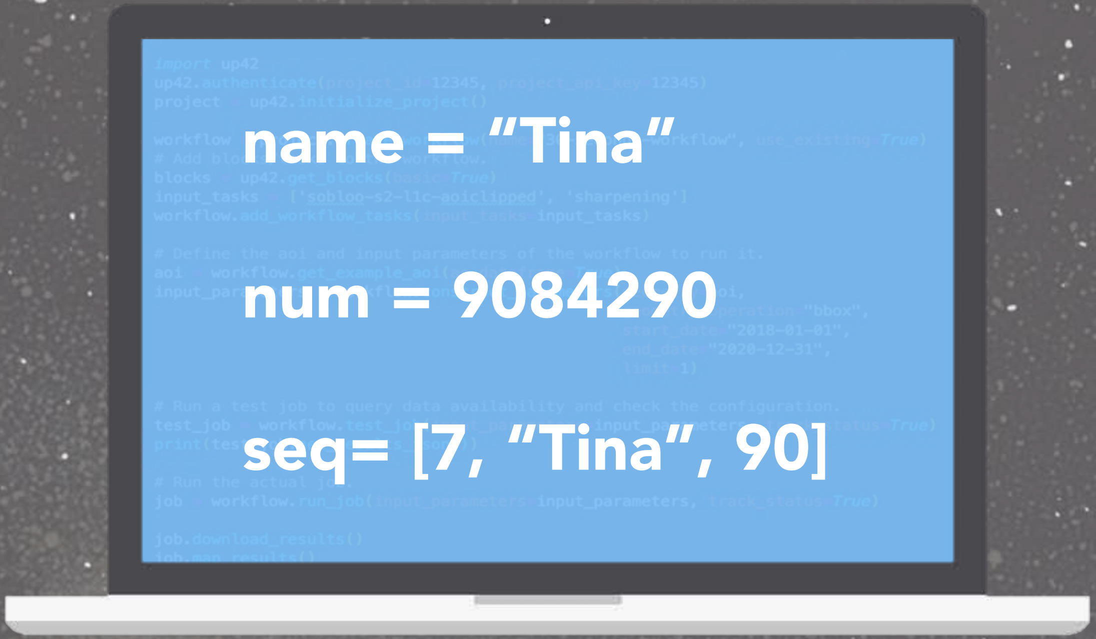
- What are variables
- How to declare a variable in Python
- Variable syntax
- How to assign a value to a variable



WHAT IS A VARIABLE?

It is a name (reference) to a memory location in the computer.

WHAT IS A VARIABLE?



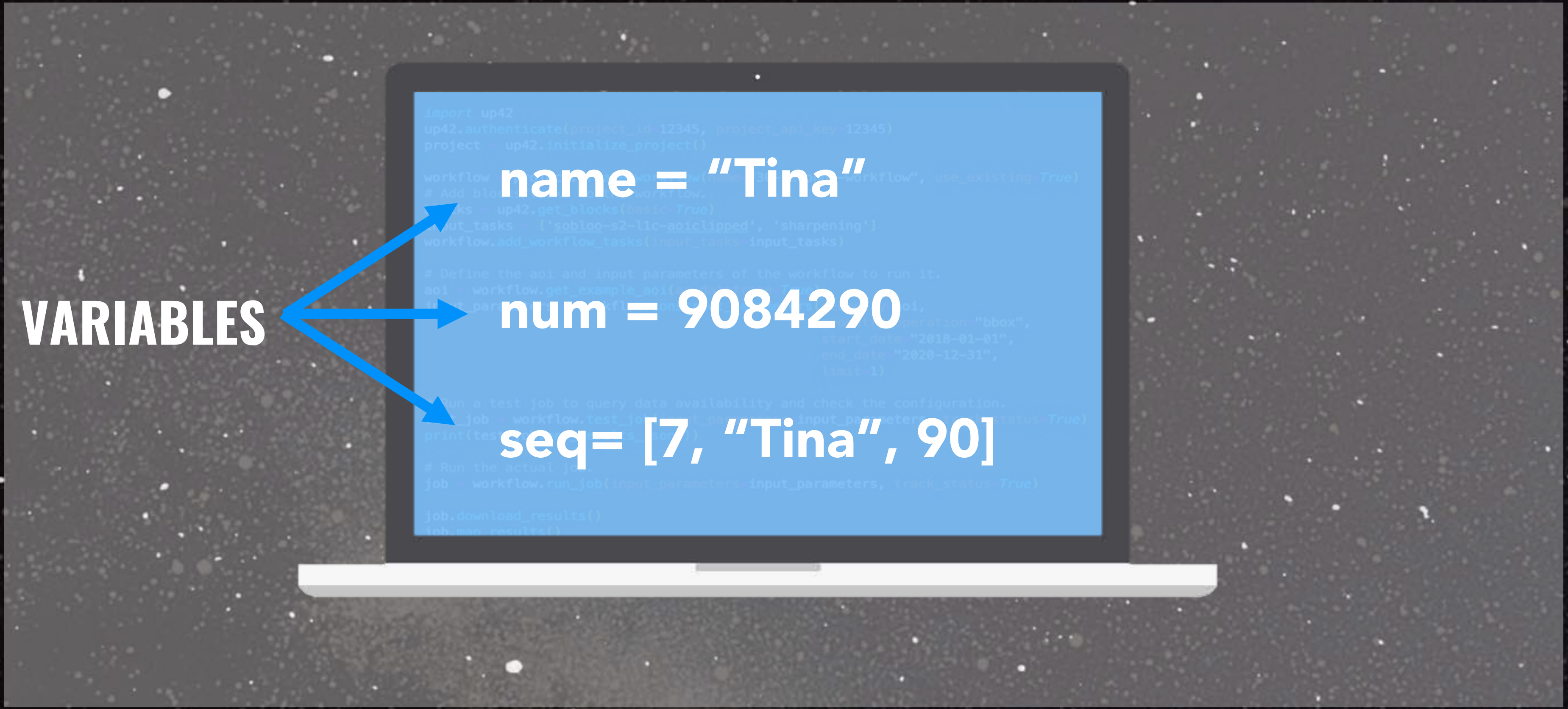
```
name = “Tina”

num = 9084290

seq= [7, “Tina”, 90]
```

It is a name (reference) to a memory location in the computer.

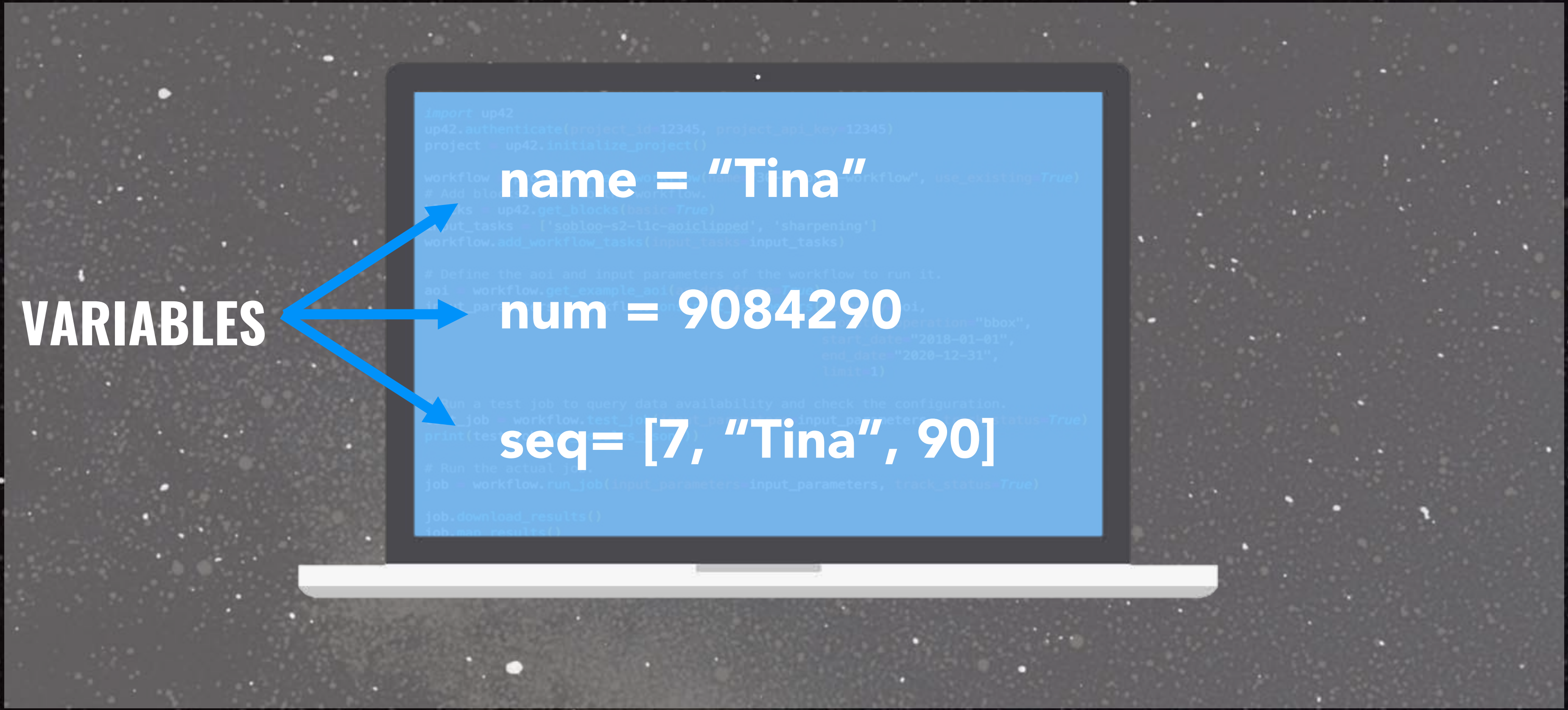
WHAT IS A VARIABLE?



```
name = “Tina”
num = 9084290
seq= [7, “Tina”, 90]
```

It is a name (reference) to a memory location in the computer.

WHAT IS A VARIABLE?

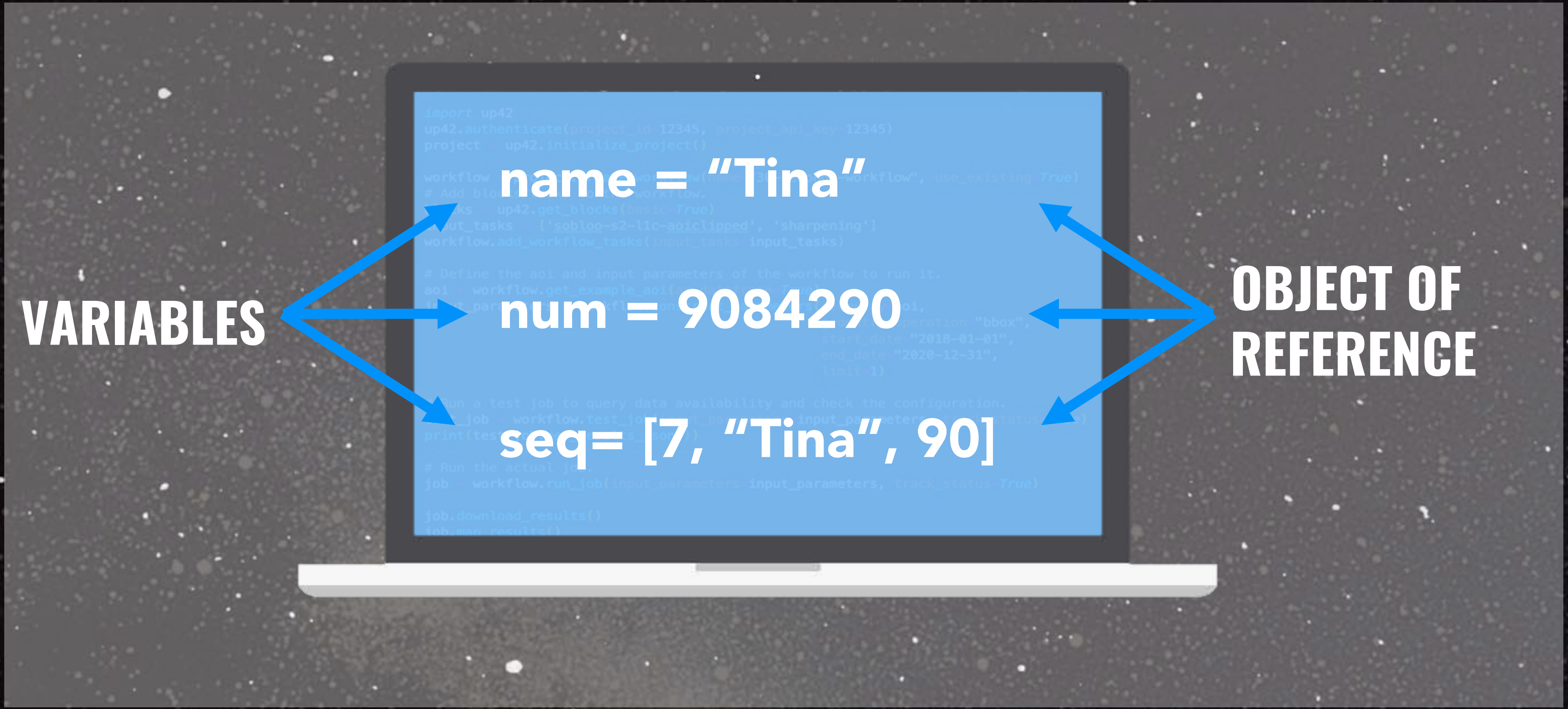


```
name = “Tina”
num = 9084290
seq= [7, “Tina”, 90]
```

It is a name (reference) to a memory location in the computer.

ALTHOUGH

WHAT IS A VARIABLE?



```
name = "Tina"
num = 9084290
seq= [7, "Tina", 90]
```

It is a name (reference) to a memory location in the computer.

ALTHOUGH

Python does not have variables. It has objects of reference.

VARIABLE

VARIABLE =

VARIABLE = OBJECT OF REFERENCE

WHAT IS IN THE OBJECT OF REFERENCE?

WHAT IS IN THE OBJECT OF REFERENCE?

DATA

WHAT IS IN THE OBJECT OF REFERENCE?

DATA

WHAT IS IN THE OBJECT OF REFERENCE?

DATA

NUMBERS

10

10+20

WHAT IS IN THE OBJECT OF REFERENCE?

DATA

NUMBERS

10

10+20

STRINGS

"tina"

'tina'

A RHYSING FOUNDATION PROGRAM SPONSORED BY UNITED AIRCRAFT TECHNOLOGIES

WHAT IS IN THE OBJECT OF REFERENCE?

DATA

NUMBERS

10

10+20

STRINGS

"tina"

'tina'

LISTS

['tina', 'turtle']

WHAT IS IN THE OBJECT OF REFERENCE?

DATA

NUMBERS
10
10+20

STRINGS
"tina"
'tina'

LISTS
['tina', 'turtle']

DICTIONARIES
{'name':'tina', 'number':1}

SYNTAX IS QUITE SIMPLE

object of reference

SYNTAX IS QUITE SIMPLE

object of reference (variable)

SYNTAX IS QUITE SIMPLE

object of reference (variable) =

SYNTAX IS QUITE SIMPLE

object of reference (variable) = value

SYNTAX IS QUITE SIMPLE

X

SYNTAX IS QUITE SIMPLE

X =

SYNTAX IS QUITE SIMPLE

```
X = 10
```

SYNTAX IS QUITE SIMPLE

```
X = 10
```

We have created an **object of type *int*** with a value 10 and an **object reference** x that refers to the *int* object.

SYNTAX IS QUITE SIMPLE

```
X = 10
```

SIMPLE MEANING: X is referring to an object with the value of 10.

EXAMPLE

$$X = 10$$

EXAMPLE

```
X = 10
print (x)
```

EXAMPLE

```
X = 10
print (x)
```

The = operator binds an object reference to an object in memory.

EXAMPLE

There are two (2) cases here:

1. If the object reference does not exist, Python creates it by the = operator.

EXAMPLE

There are two (2) cases here:

1. If the object reference does not exist, Python creates it by the = operator.

2. If the object reference already exists, it is simply re-bound to refer to the object on the right of = operator.

EXAMPLE

```
case #1
x= 10
print (x)
```

EXAMPLE

```
case #1
x= 10
print (x)
```

```
case #2
x=20
print (x)
```

```
10
20
```

EXAMPLE

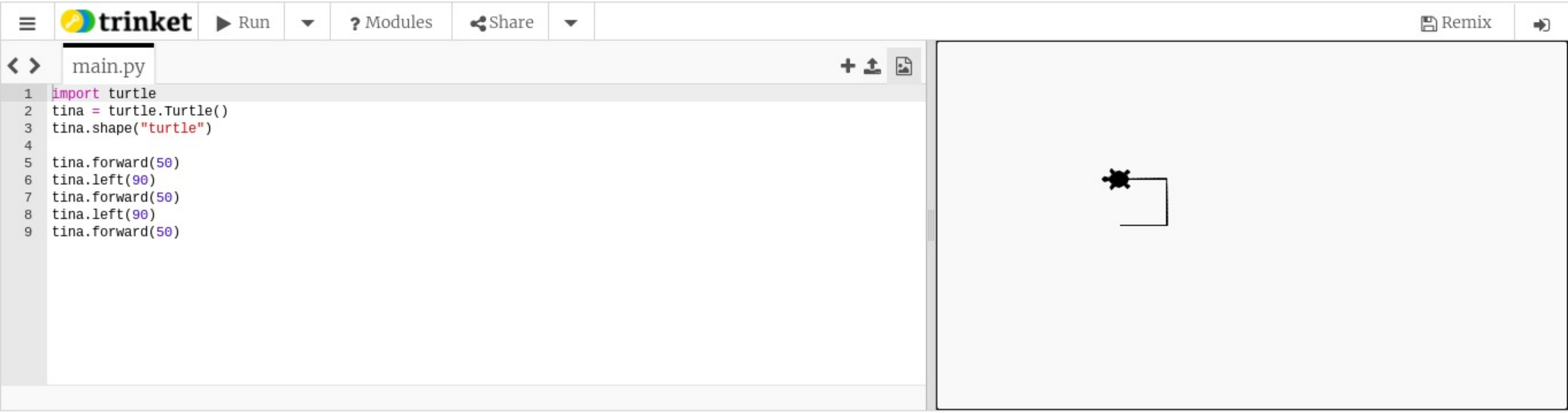
```
case #1
x= 10
print (x)
```

```
case #2
x=20
print (x)
```

```
10
20
```

EXAMPLE: TURTLE

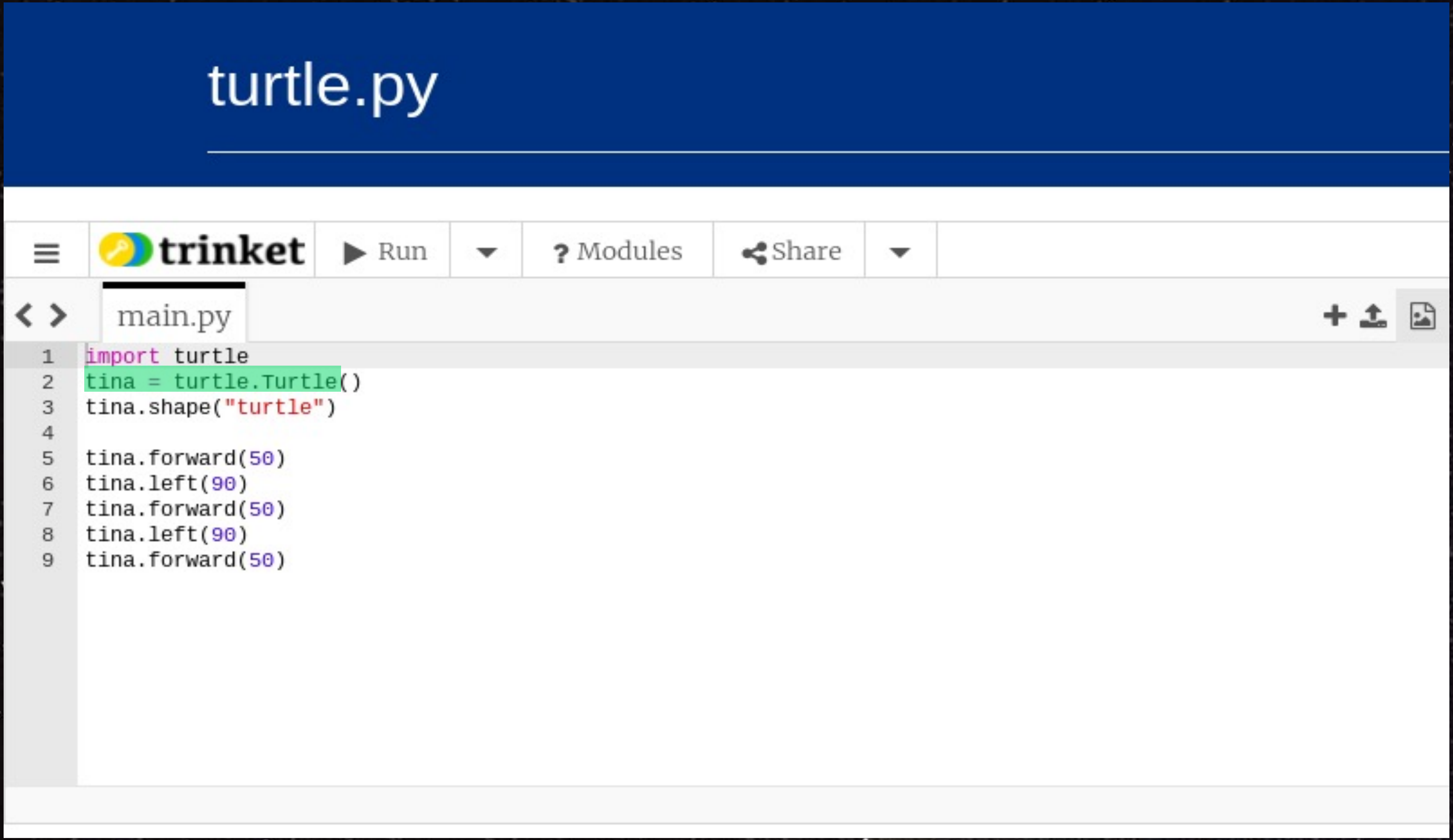
turtle.py



```python
import turtle
tina = turtle.Turtle()
tina.shape("turtle")

tina.forward(50)
tina.left(90)
tina.forward(50)
tina.left(90)
tina.forward(50)
```

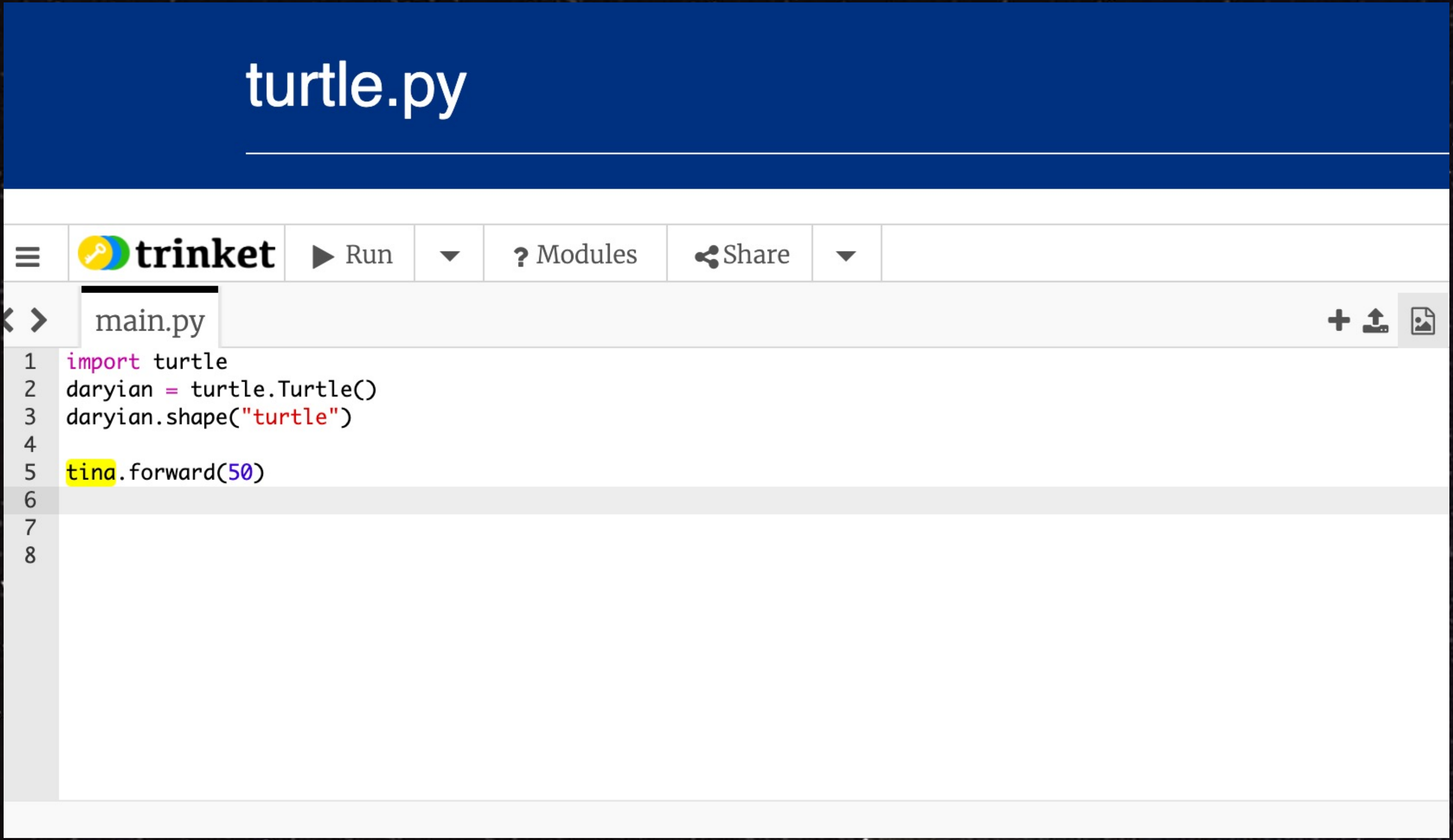
EXAMPLE: TURTLE



```python
import turtle
tina = turtle.Turtle()
tina.shape("turtle")

tina.forward(50)
tina.left(90)
tina.forward(50)
tina.left(90)
tina.forward(50)
```

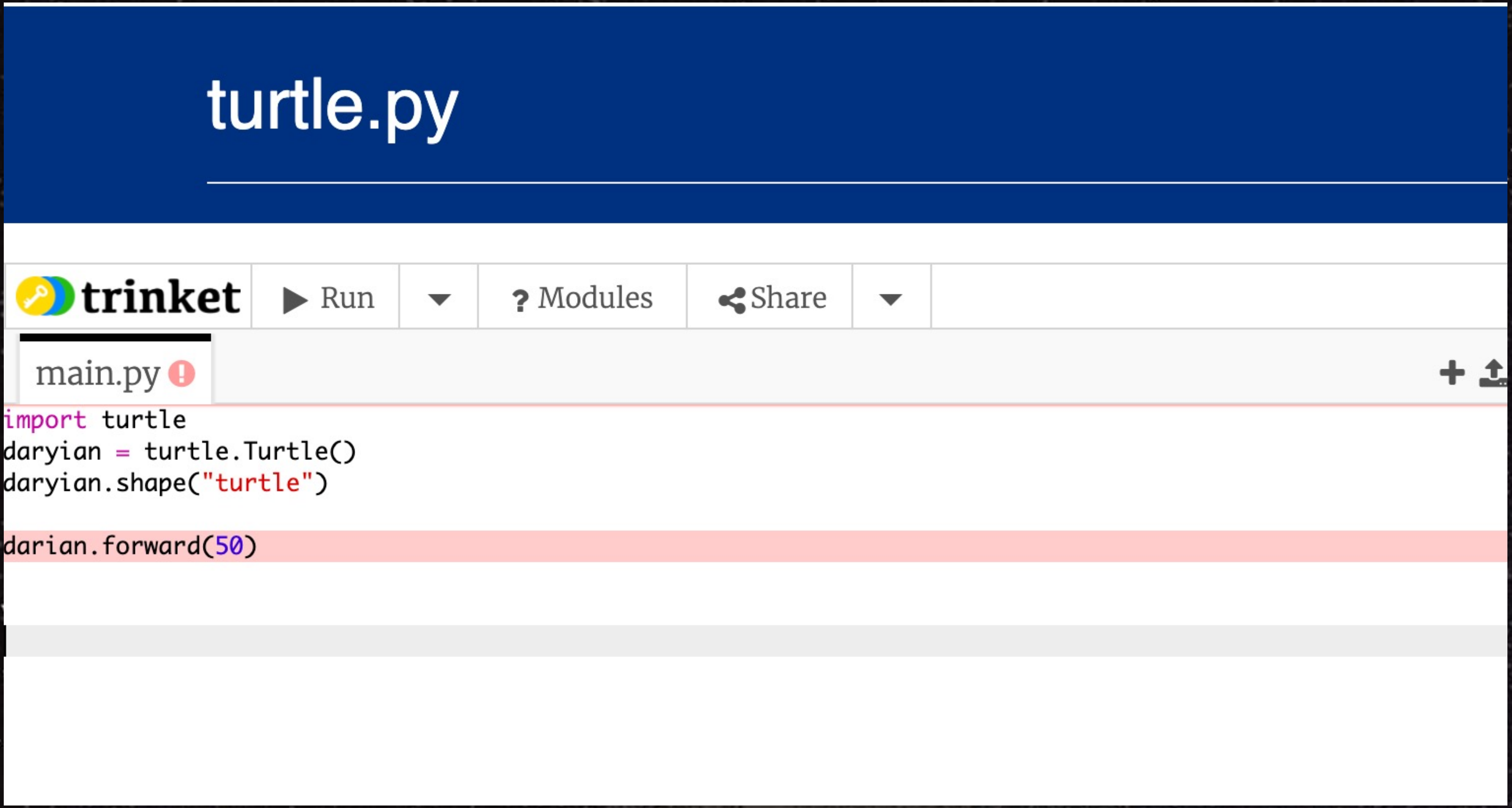
EXAMPLE: TURTLE



```python
import turtle
daryian = turtle.Turtle()
daryian.shape("turtle")

tina.forward(50)
```

EXAMPLE: TURTLE



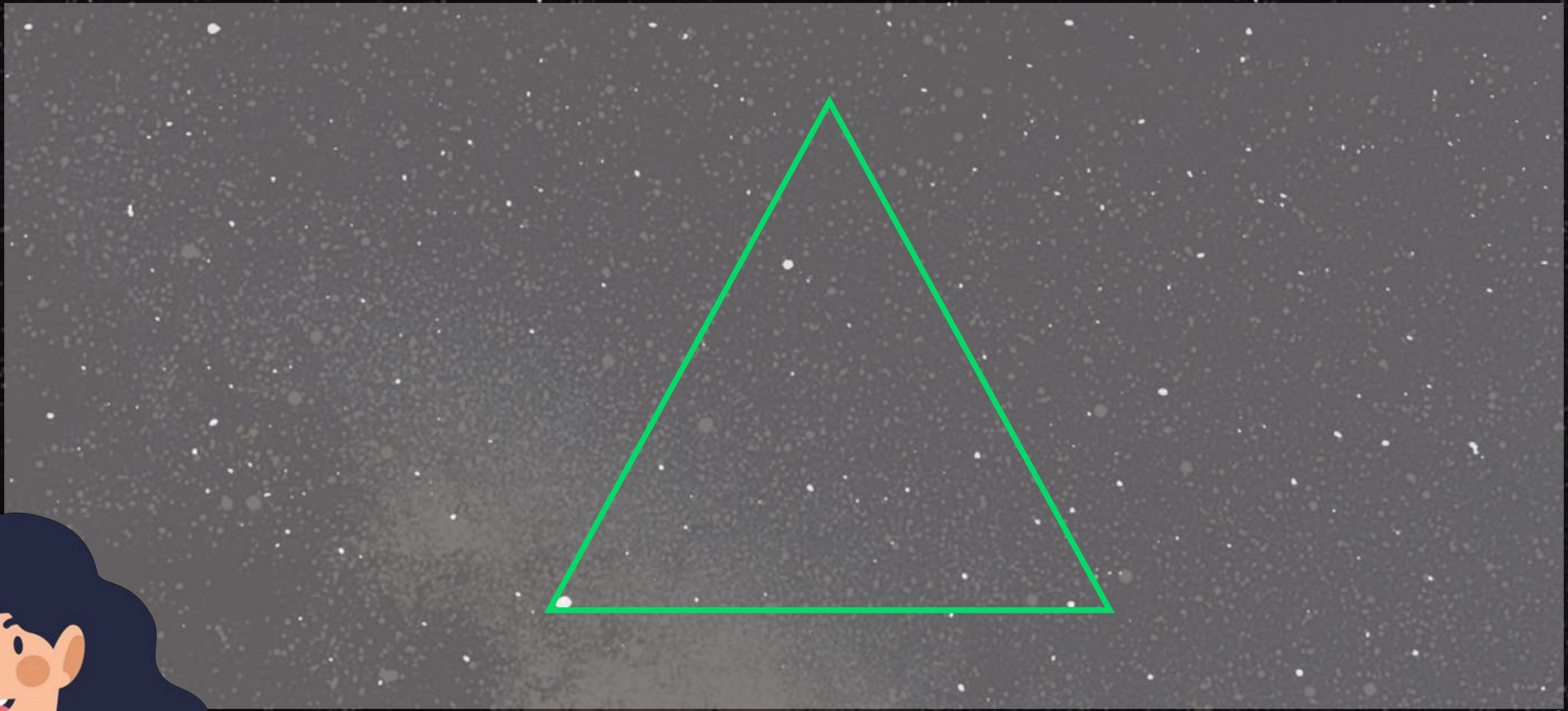
```
import turtle
daryian = turtle.Turtle()
daryian.shape("turtle")

darian.forward(50)
```

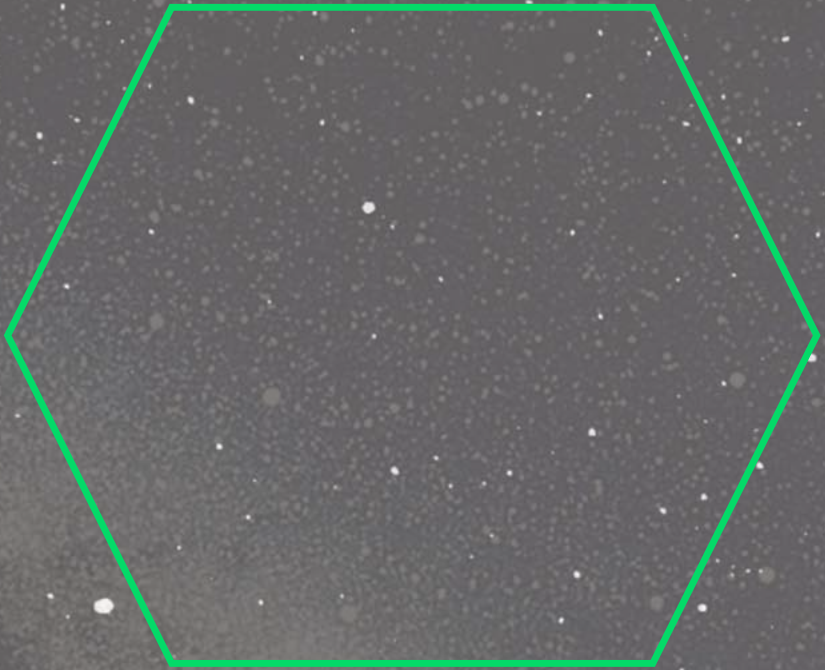
LET'S TRY IT OUT!

YOU TRY IT OUT!

YOU TRY IT OUT!



YOU TRY IT OUT!



THANK YOU FOR FLYING WITH US

A RHYSING FOUNDATION PROGRAM SPONSORED BY UNITED AIRCRAFT TECHNOLOGIES