Python Programming

Key Words

algorithm	A set of step-by-step instructions designed to solve a problem
character	A single letter, digit or punctuation mark.
data type	The format in which a variable or constant holds data, such as 'integer' or 'string'.
integer	A whole number.
iteration	The act of repeating a set of programming instructions.
program	Sequences of instructions for a computer.
selection	A programming instruction where a section of code is run only if a condition is met.
sequence	A set of programming instructions that follow on one from another.
string	A sequence of characters often stored as a variable in a computer program.
variable	A memory location within a computer program where values are stored and can be changed as the program executes.

print()

```
>>> print("Hello World")
Hello World
```

```
>>> text = "Hello"
>>> name = "Mia"
>>> print(text, name)
Hello Mia
```

comparison operators

Comparison operators are most often used in `if` statements:

```
if a > b:
    # do something
```

Operator	Name
==	equal to
!=	not equal to
>	greater than

Operator	Name
<	less than
>=	greater or equal to
<=	less or equal to

random

To use the random function, first import the random module:

```
import random
dice_number = random.randint(1,6)
```

turtle commands

Command	Arguments	Example
`forward()` or `fd()`	distance (pixels)	`forward(50)`
`back()` or `bk()`	distance (pixels)	`back(50)`
`right()` or `rt()`	angle (degrees)	`right(90)`
`left()` or `lt()`	angle (degrees)	`left(90)`
`home()`	none required (turtle to start)	
`penup()`	none required	
`pendown()`	none required	
`speed()`	10 = fast, 1 = slow, 6 = normal 0 = fast as possible	`speed(6)`
`pensize()`	line width (pixels)	`pensize(10)`
`pencolor()`	common colours	`pencolor("red")`
`shape()`	arrow, turtle, circle, square, triangle, classic	`shape("turtle")`

input()

It can only accept a single string:

```
>>> name = input("What is your name? ")
What is your name? Daniel

>>> print(name)
Daniel
```

Ask the user for an integer:

```
user_age = int(input("Enter your age: "))
```

iteration

while loops

While loops continue looping through a block of code while a test is true:

```
>>> n = 0
>>> while n < 3:
        print(n)
        n = n+1

0
1
2
>>>
```

range()

The range function takes three integer arguments:

```
range([start], [up to but not including], [steps])
```

starts from zero if omitted — required — only used with both other arguments

```
>>> for i in range(6):
        print(i, end=" ")
0 1 2 3 4 5
>>> for i in range(2,6):
        print(i, end=" ")
2 3 4 5
>>> for i in range(2,6,2):
        print(i, end=" ")
2 4
```

functions

To make your own functions use the `def` key word:

```
def add_two_numbers(a,b):
    print(a + b)
```

function name — parameters

Calling the function:

```
add_two_numbers(3,4)
7
```

arguments

"strings"

Strings can be added to and multiplied:

```
>>> my_string = "One ring to"
>>> my_string
'One ring to'

>>> my_string = my_string + " rule them all"
>>> my_string
'One ring to rule them all'

>>> my_string = my_string * 2
>>> my_string
'One ring to rule them allOne ring to rule them all'
```

Some escape sequences:

Escape sequence	What it does
`\n`	creates a line return in a string
`\t`	creates a tab style indent in a string
`\\`	allows a backslash to appear in string
`\"`	allows a speech mark to appear in a string

selection

if elif else

These control statements are used with logical operators to provide control in programs:

if

```
>>> my_number = 7
>>> if my_number > 5:
        print("My number is big.")

My number is big.
```

else

```
>>> my_number = 2
>>> if my_number > 5:
        print("My number is big.")
    else:
        print("My number is small.")

My number is small.
```

elif

```
>>> my_number = 7
>>> if my_number < 5:
        print("My number is small.")
    elif my_number < 10:
        print("My number is medium sized.")
    elif my_number < 100:
        print("My number is big.")
    else:
        print("My number is huge.")

My number is medium sized.
```

Quizizz | **Further Reading** | **Test Yourself** | **Replit** | **Quizlet** | **How to Revise**

BOHUNT EDUCATION TRUST