

Programming Techniques - Part 1

Variable

A name given to represent a value that can change during the running of the program i.e. "price"

Constant

A name given to represent a value that will not change during the running of the program i.e. "VAT_rate"

Operator

A character that represents an action for instance "+" is a mathematical operator

Input

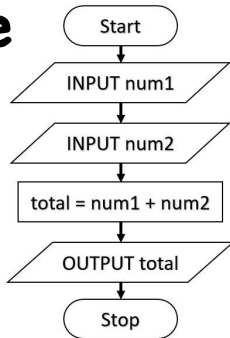
A value that is entered into the program after the program has started running

Output

A value that produced by the program and either saved or displayed to the user

Sequence

The running order of the program (usually from top to bottom unless directed to a subprogram).

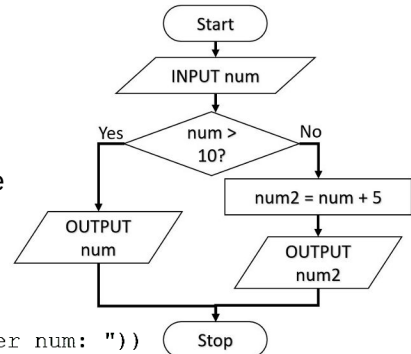


```

num1 = int(input("Enter num1: "))
num2 = int(input("Enter num2: "))
total = num1 + num2
print(total)
    
```

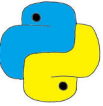
Selection

A decision is made to dictate the route that is taken, from two possible options, through the program



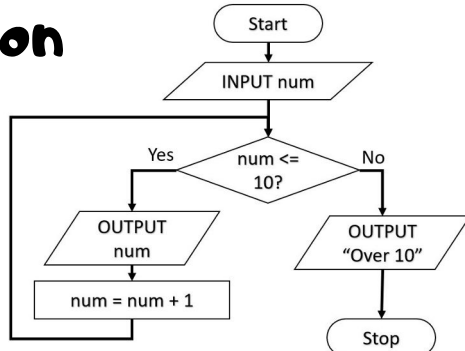
```

num = int(input("Enter num: "))
if num > 10:
    print(num)
else:
    num2 = num + 5
    print(num2)
    
```



Iteration

A loop which is repeated. This can be a condition-controlled loop or a count-controlled loop.



```

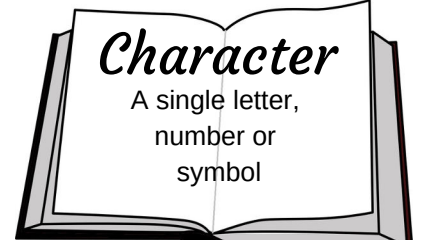
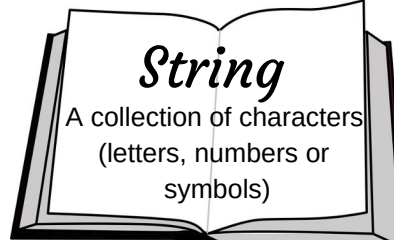
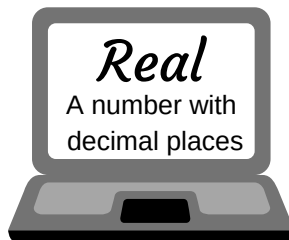
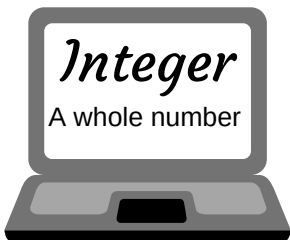
num = int(input("Enter num: "))
while num <= 10:
    print(num)
    num = num + 1
print("Over 10")
    
```



Common operators

==	>	<
Equal to	More than	Less than
!=	>=	<=
Not equal to	More than or equal to	Less than or equal to
+	-	*
Addition	Subtraction	Multiplication
/	DIV	MOD
Division	Whole number division	Remainder

(c) Nichola Wilkin Ltd 2019



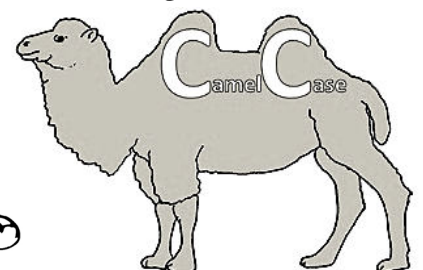
CASTING Defining or changing the types of value that a variable or constant can hold while the program is running

Integer num = int(input("Enter a whole number:"))

Real num = float(input("Enter a real number:"))

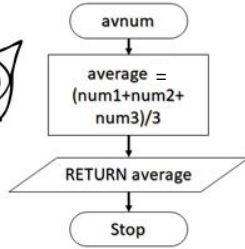
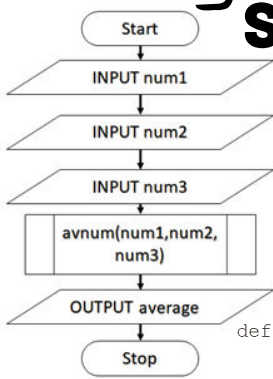
String name = input("Enter your name:")

Naming variables



Programming Techniques - Part 2

Sub programs

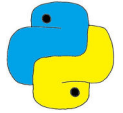


```
def avnum(num1,num2,num3):
    average = (num1+num2+num3)/3
    return average
```



```
num1 = int(input("Enter num1: "))
num2 = int(input("Enter num2: "))
num3 = int(input("Enter num3: "))
average = avnum(num1,num2,num3)
print(average)
```

Arrays



Students			
James	Anne	Connor	Tim

An 1D array is a single row of values assigned to a single name

Create 1D array `students = ["James","Anne","Connor","Tim"]`

Print single item `print(students[0])`

Print whole array `print(students)`

Find length of array `print(len(students))`

Delete single item `del students[3]`

Add item to array `students.append("Fariah")`

James	45	28
Anne	93	71
Connor	58	33

2D array

A 2D array contains multiple rows of values assigned to a single name

Create a 2D array

```
students = [["James",45,28],["Anne",93,71],["Connor",58,33]]
```

Print single item `print(students[0][1])`

Add new row of data `students.append(["Fariah",70,36])`

Changing values `students[1][2] = 74`



String Manipulation

String Length

```
name = input("Enter your name: ")
print(len(name))
```

Position of character in string

```
letterPos = name.find("k")
print(letterPos)
```

Display part of string

```
message = "Chewie, we're home."
print(message[8:13])
```

Concatenation

```
firstName = input("Enter first name: ")
surname = input("Enter surname: ")
fullName = firstName + " " + surname
```

Convert character to ASCII

```
letter = input("Enter a letter: ")
print(ord(letter))
```

Convert ASCII to character

```
num = int(input("Enter a number: "))
print(chr(num))
```

FILE HANDLING

Open and write to a text file

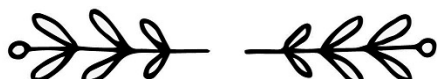
```
country = "England"
population = 53000000
file = open("countries.txt","w")
newData = country+", "+str(population)+"\n"
file.write(newData)
file.close()
```

Append to a text file

```
country = input("Enter country: ")
population = int(input("Enter population: "))
file = open("countries.txt","a")
newData = country+", "+str(population)+"\n"
file.write(newData)
file.close()
```

Read from a text file

```
file = open("countries.txt","r")
dataToRead = file.read()
print(dataToRead)
file.close()
```



SQL Databases

WRITING TO AN SQL DATABASE

```
import sqlite3
with sqlite3.connect("company.db") as db:
    cursor=db.cursor()

    cursor.execute(""" CREATE TABLE IF NOT EXISTS employees(
    id integer PRIMARY KEY,
    name text NOT NULL,
    dept text NOT NULL,
    salary integer); """)

    newID=input("Enter ID number: ")
    newName = input("Enter name: ")
    newDept = input("Enter department: ")
    newSalary = int(input("Enter salary: "))
    cursor.execute("""INSERT INTO employees(id,name,dept,salary)
    VALUES (?, ?, ?, ?) """, (newID,newName,newDept,newSalary))
    db.commit()
```

SELECTING ALL DATA FROM SQL DATABASE

```
cursor.execute("SELECT * FROM employees")
print(cursor.fetchall())
```

SELECTING FIELDS FROM SQL DATABASE

```
cursor.execute("SELECT id,name,salary FROM employees")
print(cursor.fetchall())
```

SELECTING DATA THAT MEETS A CONDITION

```
cursor.execute("SELECT * FROM employees WHERE salary>20000")
print(cursor.fetchall())
```

SELECTING DATA USING A WILDCARD

```
cursor.execute("SELECT name FROM employees WHERE name LIKE 'M%'")
print(cursor.fetchall())
```

Programming Techniques

Revise it



Problem 1

Ask the user to enter some names and ages and store them in a 2D array. Once they have created 6 records, allow them to pick a number between 0 and 5 and change the age.

Problem 2

Using subprograms, create a program that will allow users to either add a name to a text file, or display the contents of the text file.

The best way to learn a programming language is to have a go at solving problems. Try to create working, user friendly solution for ALL the following problems.



Problem 3

Create an SQL database to keep test scores, that stores student's name and their test score. Run it several times to build the list then only show then show only the students who passed the test (i.e scored 50 or more).

(c) Nichola Wilkin Ltd 2019

Example answers

There are many ways of solving the problems, here are some possible solutions to help you, in case you get stuck, and to compare against your own solution.

```
names=[]
for i in range(0,6):
    name = input("Enter name: ")
    age = int(input("Enter age: "))
    newrecord = [name,age]
    names.append(newrecord)
selection = int(input("Enter number between 0 and 5: "))
newage = int(input("Enter new age: "))
names[selection][1] = [newage]
print(names)
```

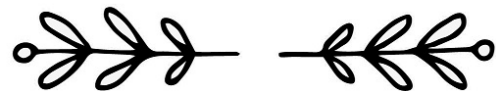
Problem 1

```
def addname():
    newname = input("Enter name: ")
    file = open("name.txt","a")
    file.write(newname+"\n")
    file.close()

def displayfile():
    file = open("name.txt","r")
    datatoread = file.read()
    print(datatoread)
    file.close()

print("***MENU**")
print()
print("1) Add name")
print("2) View names")
selection = int(input("Enter number: "))
if selection == 1:
    addname()
elif selection == 2:
    displayfile()
else:
    print("Invalid selection")
```

Problem 2



Problem 3

```
import sqlite3
with sqlite3.connect("test.db") as db:
    cursor=db.cursor()

    cursor.execute(""" CREATE TABLE IF NOT EXISTS scores(
    name text PRIMARY KEY,
    score integer); """)

    newname=input("Enter name: ")
    newscore=int(input("Enter score: "))
    cursor.execute("""INSERT INTO scores(name,score)
    VALUES (?,?) """, (newname,newscore))
    db.commit()

    cursor.execute("SELECT * FROM scores WHERE score >= 50")
    print(cursor.fetchall())
```

Programming Techniques - Part 1

Variable

A name given to represent a value that can change during the running of the program i.e. "price"

Constant

A name given to represent a value that will not change during the running of the program i.e. "VAT_rate"

Operator

A character that represents an action for instance "+" is a mathematical operator

Input

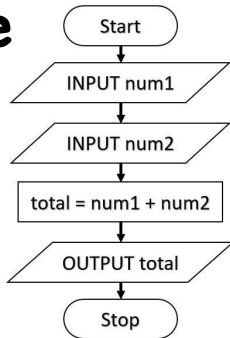
A value that is entered into the program after the program has started running

Output

A value that produced by the program and either saved or displayed to the user

Sequence

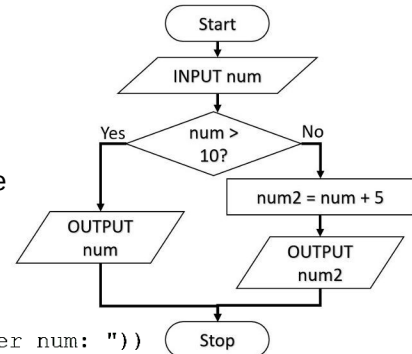
The running order of the program (usually from top to bottom unless directed to a subprogram).



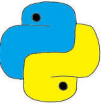
```
num1 = int(input("Enter num1: "))
num2 = int(input("Enter num2: "))
total = num1 + num2
print(total)
```

Selection

A decision is made to dictate the route that is taken, from two possible options, through the program

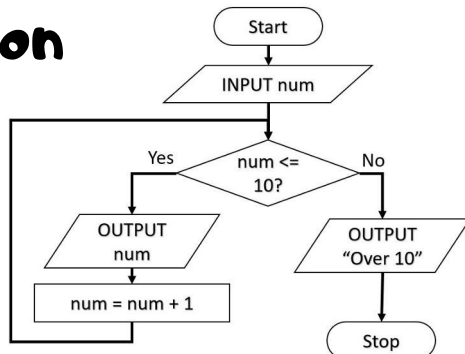


```
num = int(input("Enter num: "))
if num > 10:
    print(num)
else:
    num2 = num + 5
    print(num2)
```



Iteration

A loop which is repeated. This can be a condition-controlled loop or a count-controlled loop.



```
num = int(input("Enter num: "))
while num <= 10:
    print(num)
    num = num + 1
print("Over 10")
```



Common operators

==	>	<
Equal to	More than	Less than
!=	>=	<=
Not equal to	More than or equal to	Less than or equal to
+	-	*
Addition	Subtraction	Multiplication
/	DIV	MOD
Division	Whole number division	Remainder

(c) Nichola Wilkin Ltd 2019



Integer
A whole number

Real
A number with decimal places

String
A collection of characters (letters, numbers or symbols)

Character
A single letter, number or symbol

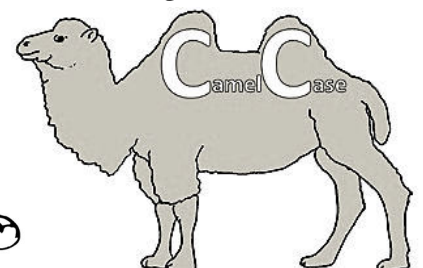
CASTING Defining or changing the types of value that a variable or constant can hold while the program is running

Integer num = int(input("Enter a whole number:"))

Real num = float(input("Enter a real number:"))

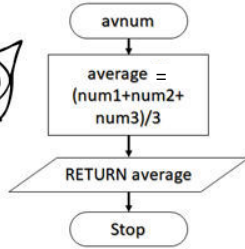
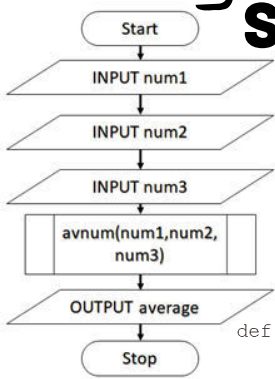
String name = input("Enter your name:")

Naming variables



Programming Techniques - Part 2

Sub programs

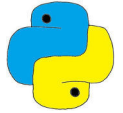


```
def avnum(num1, num2, num3):
    average = (num1+num2+num3)/3
    return average
```



```
num1 = int(input("Enter num1: "))
num2 = int(input("Enter num2: "))
num3 = int(input("Enter num3: "))
average = avnum(num1, num2, num3)
print(average)
```

Arrays



Students			
James	Anne	Connor	Tim

An 1D array is a single row of values assigned to a single name

Create 1D array `students = ["James", "Anne", "Connor", "Tim"]`

Print single item `print(students[0])`

Print whole array `print(students)`

Find length of array `print(len(students))`

Delete single item `del students[3]`

Add item to array `students.append("Fariah")`

James	45	28
Anne	93	71
Connor	58	33

2D array

A 2D array contains multiple rows of values assigned to a single name

Create a 2D array

```
students = [["James", 45, 28], ["Anne", 93, 71], ["Connor", 58, 33]]
```

Print single item `print(students[0][1])`

Add new row of data `students.append(["Fariah", 70, 36])`

Changing values `students[1][2] = 74`



String Manipulation

String Length

```
name = input("Enter your name: ")
print(len(name))
```

Position of character in string

```
letterPos = name.find("k")
print(letterPos)
```

Display part of string

```
message = "Chewie, we're home."
print(message[8:13])
```

Concatenation

```
firstName = input("Enter first name: ")
surname = input("Enter surname: ")
fullName = firstName + " " + surname
```

Convert character to ASCII

```
letter = input("Enter a letter: ")
print(ord(letter))
```

Convert ASCII to character

```
num = int(input("Enter a number: "))
print(chr(num))
```

FILE HANDLING

Open and write to a text file

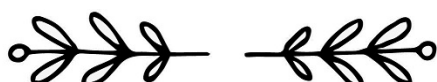
```
country = "England"
population = 53000000
file = open("countries.txt", "w")
newData = country + ", " + str(population) + "\n"
file.write(newData)
file.close()
```

Append to a text file

```
country = input("Enter country: ")
population = int(input("Enter population: "))
file = open("countries.txt", "a")
newData = country + ", " + str(population) + "\n"
file.write(newData)
file.close()
```

Read from a text file

```
file = open("countries.txt", "r")
dataToRead = file.read()
print(dataToRead)
file.close()
```



SQL Databases

WRITING TO AN SQL DATABASE

```
import sqlite3
with sqlite3.connect("company.db") as db:
    cursor=db.cursor()
```

```
cursor.execute(""" CREATE TABLE IF NOT EXISTS employees(
id integer PRIMARY KEY,
name text NOT NULL,
dept text NOT NULL,
salary integer); """)
```

```
newID=input("Enter ID number: ")
newName = input("Enter name: ")
newDept = input("Enter department: ")
newSalary = int(input("Enter salary: "))
cursor.execute("""INSERT INTO employees(id, name, dept, salary)
VALUES (?, ?, ?, ?) """, (newID, newName, newDept, newSalary))
db.commit()
```

SELECTING ALL DATA FROM SQL DATABASE

```
cursor.execute("SELECT * FROM employees")
print(cursor.fetchall())
```

SELECTING FIELDS FROM SQL DATABASE

```
cursor.execute("SELECT id, name, salary FROM employees")
print(cursor.fetchall())
```

SELECTING DATA THAT MEETS A CONDITION

```
cursor.execute("SELECT * FROM employees WHERE salary > 20000")
print(cursor.fetchall())
```

SELECTING DATA USING A WILDCARD

```
cursor.execute("SELECT name FROM employees WHERE name LIKE 'M%'")
print(cursor.fetchall())
```

Programming Techniques

Revise it

Problem 1

Ask the user to enter some names and ages and store them in a 2D array. Once they have created 6 records, allow them to pick a number between 0 and 5 and change the age.

Problem 2

Using subprograms, create a program that will allow users to either add a name to a text file, or display the contents of the text file.

The best way to learn a programming language is to have a go at solving problems. Try to create working, user friendly solution for ALL the following problems.



Problem 3

Create an SQL database to keep test scores, that stores student's name and their test score. Run it several times to build the list then only show then show only the students who passed the test (i.e scored 50 or more).

(c) Nichola Wilkin Ltd 2019

Example answers

There are many ways of solving the problems, here are some possible solutions to help you, in case you get stuck, and to compare against your own solution.

```
names=[]
for i in range(0,6):
    name = input("Enter name: ")
    age = int(input("Enter age: "))
    newrecord = [name,age]
    names.append(newrecord)
selection = int(input("Enter number between 0 and 5: "))
newage = int(input("Enter new age: "))
names[selection][1] = [newage]
print(names)
```

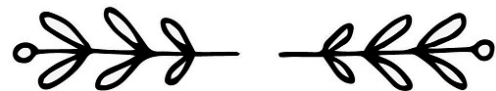
Problem 1

```
def addname():
    newname = input("Enter name: ")
    file = open("name.txt","a")
    file.write(newname+"\n")
    file.close()

def displayfile():
    file = open("name.txt","r")
    datatoread = file.read()
    print(datatoread)
    file.close()

print("***MENU**")
print()
print("1) Add name")
print("2) View names")
selection = int(input("Enter number: "))
if selection == 1:
    addname()
elif selection == 2:
    displayfile()
else:
    print("Invalid selection")
```

Problem 2



Problem 3

```
import sqlite3
with sqlite3.connect("test.db") as db:
    cursor=db.cursor()

    cursor.execute(""" CREATE TABLE IF NOT EXISTS scores(
name text PRIMARY KEY,
score integer); """)

    newname=input("Enter name: ")
    newscore=int(input("Enter score: "))
    cursor.execute("""INSERT INTO scores(name,score)
VALUES (?,?) """, (newname,newscore))
    db.commit()

    cursor.execute("SELECT * FROM scores WHERE score >= 50")
    print(cursor.fetchall())
```