IHF: CODE

PYTHON – SESSION 2

REVIEW

TEXT EDITOR

<USER>.CODE.IHF.APPS.CLOUD-OPS.CO.UK

RUNNING A PYTHON SCRIPT

RUNNING A PYTHON SCRIPT

```
$ python <file_name>.py
```

```
$ python hello_world.py
Hello, World!
```

VARIABLES

VARIABLES

```
name = "Charlie"
age = 27
left_to_pay = 29.99
has_paid = False
```

DATA TYPES

DATA TYPES

TYPE	EXAMPLE	
String	"Alice"	Characters surrounded by quotes
Integer	13	Whole number
Float	3.99	Decimal number
Boolean	True	True or False
None	None	Absence of value

NUMERICAL OPERATORS

NUMERICAL OPERATORS

```
print(1 + 2) # Addition / Concatenation
print(5 - 3) # Subtraction
print(3 * 7) # Multiplication
print(49 / 7) # Division
print(4 ** 2) # Exponent
print(10 % 3) # Modulus (Remainder)
```

COMMENTS

COMMENTS

```
# The total including VAT
total = sub_total + vat

has_paid = False # If the user has paid or not
```

CASTING

CASTING

```
x = int(8.3)  # x will be 8
y = float(9)  # y will be 9.0
z = str(10.0) # z will be '10.0'
```

INDEX

INDEX

C	H	A	R	L	I	E
0	1	2	3	4	5	6

```
name = "CHARLIE"
print(name[0]) # Prints 'C'
print(name[1]) # Prints 'H'
```

INPUT

INPUT

```
name = input("What's your name? ")
print("Hello " + name)

age = int(input("How old are you? "))
age_in_10_years = age + 10
print("In 10 years you will be " + str(age_in_10_years))
```

QUESTIONS?

CONDITIONALS

IF

IF

```
if True:
    print("This is always shown")

if False:
    print("This is never shown")
```

IF

```
if <expression>:
    # Will only be run if the expression is True
    <code>
```

INDENTING

INDENTING

```
name = "Alice"
if name == "Alice":
    print("Hello Alice")
```

COMPARATORS

COMPARATOR	DESCRIPTION	EXAMPLE
==	Equals	"Alice" == "Alice"
!=	Does not equal	"Bob" != "Charlie"
<	Less than	4 < 10
>	Greater than	12 > 8
<=	Less than or equal to	7 <= 7
>=	Greater than or equal to	8 >= 5

COMPARATORS

```
name = "Alice"
if name == "Alice":
    print("Hello Alice")

if name != "Charlie":
    print("You are not Charlie")

age = 24
if age >= 21:
    print("You are 21 or over")
```

ELSE

ELSE

```
if 1 == 1:
    print("Yes")
else:
    print("No")
```

ELSE

```
if <expression>:
    # Will only be run if the expression is True
    <code>
else:
    # Will only be run if the expression is False
    <code>
```

ELSE

```
name = "Alice"
if name == "Alice":
    print("Hello Alice")
else:
    print("You are not Alice")

age = 24
if age >= 21:
    print("You are 21 or over")
else:
    print("You are 20 or younger")
```

CODING TIME

SECTION A

ELIF

ELIF

```
name = "Bob"
if name == "Alice":
    print("Hello Alice")
elif name == "Bob":
    print("Hello Bob")
else:
    print("You must be Charlie")
```

ELIF

```
age = int(input("How old are you? "))
if age <= 13:
    print("You are 13 or younger")
elif age < 18:
    print("You are between 14 and 17")
elif age == 18:
    print("You are 18")
else:
    print("You are 19 or over")
```

AND, OR, NOT

AND, OR, NOT

```
if age > 12 and age < 20:
    print("You are a teenager")

if age < 13 or age > 19:
    print("You are not a teenager")

if not (age > 12 and age < 20):
    print("You are not a teenager")
```

CODING TIME

SECTION B

LIST

LIST

```
names = ["Alice", "Bob", "Charlie"]
print(names[0]) # Alice
print(names[1]) # Bob
print(names[2]) # Charlie
```

LIST – APPEND

```
names = ["Alice", "Bob", "Charlie"]

# Append a new value to the end of the list
names.append("Dave")

print(names) # ["Alice", "Bob", "Charlie", "Dave"]
```

LIST – CHANGE

```
names = ["Alice", "Bob", "Charlie"]

# Change the value stored in index 2
names[2] = "Chris"

print(names) # ["Alice", "Bob", "Chris"]
```

LIST – DELETE

```
names = ["Alice", "Bob", "Charlie"]

# Delete the item at index 1
del(names[1])

print(names) # ["Alice", "Charlie"]
```

LIST – IN

```
names = ["Alice", "Bob", "Charlie"]

# Check to see if "Eve" is in the list
if "Eve" in names:
    print("Eve is here")
else:
    print("Eve isn't here")
```

LIST – LENGTH

```
names = ["Alice", "Bob", "Charlie"]

# Get the number of items in the list
print(len(names)) # 3
```

LIST – SORT

```
names = ["Charlie", "Alice", "Bob"]

# Sort the items in the list alphabetically
names.sort()

print(names) # ["Alice", "Bob", "Charlie"]
```

CODING TIME

SECTION C

FOR LOOPS

FOR LOOPS

```
names = ["Alice", "Bob", "Charlie"]

for person in names:
    print(person)

# Alice
# Bob
# Charlie
```

FOR LOOPS

```
for <item> in <items>:
    # Runs once for each item in items
    <code>
```

FOR LOOPS

```
for my_number in range(5):
    print(my_number)

# 0
# 1
# 2
# 3
# 4
```

RANGES

RANGES

```
range(10)
#[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

range(1, 5)
#[1, 2, 3, 4]

range(2000, 2020, 4)
#[2000, 2004, 2008, 2012, 2016]
```

CODING TIME

SECTION D

EXERCISES

Finish off any exercises you did not complete in the session

FURTHER HELP

DL-UKIHFCODE@KPMG.CO.UK