IHF: CODE

PYTHON – SESSION 5

REVIEW

FUNCTIONS

FUNCTIONS – CREATE

```
def hello_world():
    print("Hello World!")
```

FUNCTIONS – CREATE

```
def <function_name>():
    <your code here>
```

FUNCTIONS – CALL

```
def hello_world():
    print("Hello World!")

hello_world()
```

FUNCTIONS – PARAMETERS

```
def hello(name):
    print("Hello, " + name + "!")

hello("Alice")
hello("Bob")
hello("Charlie")
```

FUNCTIONS – PARAMETERS

```
def hello(name, age):
    print("Hello my name is " + name)
    print("I'm " + str(age) + " years old")

    age_in_10_years = age + 10
    print("In 10 years time I will be " + str(age_in_10_years))

hello("Alice", 22)
hello("Bob", 34)
hello("Charlie", 17)
```

FUNCTIONS – PARAMETERS

```
def area(x, y, z):
    print("The area is " + str(x * y * z))

area(12, 3, 4)
area(6, 14, 10)
```

FUNCTIONS – PARAMETERS

```
def <function_name>(<param_1>, <param_2>, ...):
    <your code here>
```

FUNCTIONS – RETURNING

```
def area(x, y, z):
    return x * y * z

cube1 = area(12, 3, 4)
cube2 = area(6, 14, 10)
```

FUNCTIONS – RETURNING

```
def <function_name>(<param_1>, <param_2>, ...):
    <your code here>

    return <value>
```

FUNCTIONS – SINGLE JOB

```
def hello(name, age):
    print("Hello my name is " + name)
    print("I'm " + str(age) + " years old")
    print("In 10 years time I will be " + str(age_in_x_years(age, 10)))

def age_in_x_years(age, years):
    return age + years

hello("Alice", 22)
hello("Bob", 34)
hello("Charlie", 17)
```

FUNCTIONS – RECURSION

```
def calc_factorial(x):
    if x == 1:
        return 1
    else:
        return (x * calc_factorial(x - 1))

num = 4
print("The factorial of " + num + " is " + str(calc_factorial(num)))
```

FUNCTIONS – RECURSION

```
def calc_factorial(x):
    if x == 1:
        return 1
    else:
        return (x * calc_factorial(x - 1))

# calc_factorial(4)                   # 1st call with 4
# 4 * calc_factorial(3)               # 2nd call with 3
# 4 * 3 * calc_factorial(2)           # 3rd call with 2
# 4 * 3 * 2 * calc_factorial(1)       # 4th call with 1
# 4 * 3 * 2 * 1                       # return from 4th call as number=1
# 4 * 3 * 2                           # return from 3rd call
# 4 * 6                               # return from 2nd call
# 24                                  # return from 1st call
```

QUESTIONS?

FILES

FILES – OPEN

```
f = open("<file>.txt", "r")
```

FILES – READ

```
f = open("<file>.txt", "r")
print(f.read())
```

FILES – READ

```
f = open("<file>.txt", "r")
for x in f:
  print(x)
```

SECTION A

CD KPMG-PYTHON-COURSE/

FILES – HANDLING

VALUE	ACTION	DESCRIPTION
"r"	Read	Opens a file for reading, error if the file does not exist
"a"	Append	Opens a file for appending, creates the file if it does not exist
"w"	Write	Opens a file for writing, creates the file if it does not exist
"x"	Create	Creates the specified file, returns an error if the file exists

FILES – WRITE

```
f = open("example.txt", "w")
f.write("Hello World")
f.close()
```

FILES – WRITE

```
f = open("example.txt", "w")
f.write("Hello World")
f.close()

f = open("example.txt", "a")
f.write("It's nice to be here")
f.close()
```

FILES – WRITE

```
f = open("names.txt", "a")

name = True
while name:
    name = input("Enter a name: ")
    f.write(name + "\n")

f.close()
```

CODING TIME

SECTION B

EXERCISES

Finish off any exercises you did not complete in the session

FURTHER HELP

DL-UKIHFCODE@KPMG.CO.UK