COMPSCI 111 / 111G

Mastering Cyberspace:
An introduction to practical computing

Programming with Python

Conditions

Revision

Ask the user to enter a number of centimetres. The program should print out the number of metres and centimeters that are equal to the input.

```
#Author: Andrew Luxton-Reilly
#Date: 7/05/06

#Ask the user to enter the number of centimetres
cm = input("Please enter the number of centimetres: ")

#Calculate the number of metres and centimetres
m = cm / 100
cm = cm % 100

#Print the output to the user
print m, "metres and", cm, "centimeters"
```

Exercise

`if` statements

```
if <condition>:
    statements to execute if condition is true
else:
    statements to execute if condition is false
```

A colon appears in two places

- after the <condition>
- after the keyword `else`

The <condition> should be a boolean expression

- Evaluates to either true or false

Statements to be executed are organised as blocks

- Statements should be indented to the same amount
- Any number of statements can be executed in a block

Comparison operators

Used to compare two things

- Evaluates to a boolean value
- Commonly used in a condition

Meaning	Operator	Example
Less than	`<`	`a < b`
Less than or equal to	`<=`	`a <= b`
Greater than	`>`	`a > b`
Greater than or equal to	`>=`	`a >= b`
Equal to	`==`	`a == b`
Not equal to	`!=`	`a != b`

Example

Write a program that asks the user to enter their PIN. If the PIN that the user enters is equal to 1234, then the program should respond with "Correct". If the user enters anything else, it should respond with "Wrong PIN".

```
#Author: Andrew Luxton-Reilly
#Date: 8/05/07

#Ask the user to enter a PIN
pin = input("Please enter your PIN: ")

#Respond with "Correct" or "Wrong PIN"
if (pin == 1234):
    print "Correct"
else:
    print "Wrong PIN"
```

Exercise

Example

Write a program that asks the user to enter a number. The program should determine if the number is odd or even, and print out an appropriate message.

```
#Author: Andrew Luxton-Reilly
#Date: 7/05/06

#Ask the user to enter a number
number = input("Please enter a number: ")

#Determine if the number is odd or even
if (number % 2 == 0):
    print "You entered the number", number
    print "That number is even"
else:
    print "You entered the number", number
    print "That number is odd"
```

Exercise

Write a program that asks the user to enter their age, and then prints out a ticket based on their age:

- Child tickets (below the age of 12) cost $5.00
- Adult tickets (age 12 or above) cost $9.99

The tickets should look like:

```
************        *************
Child Ticket        Adult Ticket
Price: $5.00        Price: $10.00
************        *************
```

Logical operators

Used to combine boolean expressions

- And
- Or
- Not

Meaning	Operator	Example
Logical AND	`and`	`number >= 1 and number <= 2`
Logical OR	`or`	`number == 1 or number == 2`
Logical NOT	`not`	`not( number < 1 or number > 2)`

Example

Write a program that asks the user to enter a number between 1 and 10 (inclusive). The program will print out "Correct" if the number is in the range and "Incorrect" if the number is outside the range.

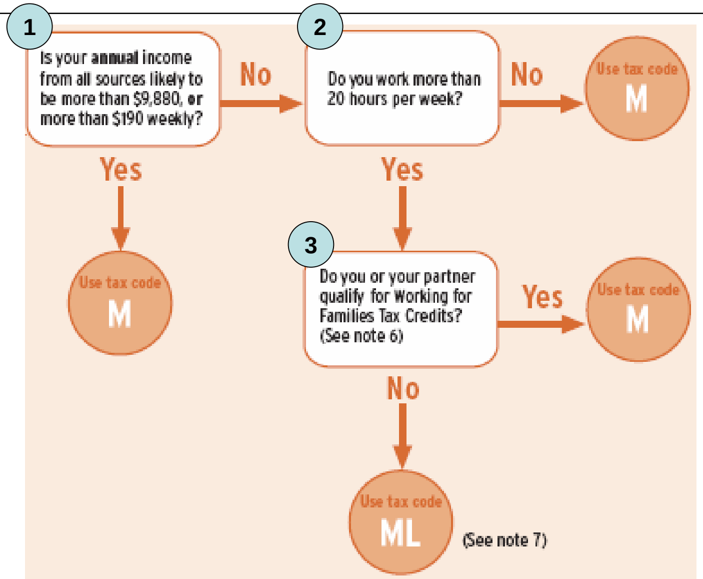
```
#Author: Andrew Luxton-Reilly
#Date: 7/05/06

#Ask the user to enter a number
number = input("Please enter a number (1-10): ")

#Determine if the number is within the range
if (number >= 1 and number <= 10):
    print "Correct"
else:
    print "Incorrect"
```

Exercise

Example: Work out tax code


1
Is your annual income from all sources likely to be more than $9,880, or more than $190 weekly?
No
2
Do you work more than 20 hours per week?
No
Use tax code M
Yes
Use tax code M
Yes
3
Do you or your partner qualify for Working for Families Tax Credits? (See note 6)
Yes
Use tax code M
No
Use tax code ML
(See note 7)

Logical structure

```
If answer to first question is yes
    Use tax code M
Else
    If answer to second question is no
        Use tax code M
    Else
        if answer to third question is yes
            Use tax code M
        Else
            Use tax code ML
```

Writing the python code

Develop the program in small steps and test frequently

```
#Author: Andrew Luxton-Reilly
#Date: 01/05/07

#Ask the user the first question
response = raw_input("Is your annual income more than
9880 (i.e. 190 per week)? (y/n) ")

#Determine the code based on the first question
if (response == "y"):
    print "Your tax code is M"
else:
    print "We still need to do more calculations"
```

Writing the python code

Develop the program in small steps and test frequently

```
#Author: Andrew Luxton-Reilly
#Date: 01/05/07

#Ask the user the first question
response = raw_input("Income > 9880 (y/n) ")

#Determine the code based on the first question
if (response == "y"):
    print "Your tax code is M"
else:
    #Test the second condition
    response = raw_input("Work > 20 hours per week?")
    if (response == "n"):
        print "Your tax code is M"
    else:
        print "Still more calculation required"
```

Writing the python code

Develop the program in small steps and test frequently

```
response = raw_input("Income > 9880 (y/n) ")

if (response == "y"):
    print "Your tax code is M"
else:
    response = raw_input("Work > 20 hours per week? ")
    if (response == "n"):
        print "Your tax code is M"
    else:
        response = raw_input("Qualify for WFF tax credit? ")
        if (response == "y"):
            print "Your tax code is M"
        else:
            print "Your tax code is ML"
```