

Advanced Programming

Key Words

ASCII	American Standard Code for Information Interchange. A 7-bit character set used for representing English keyboard characters.
database	A data store designed in an organised way, making it easier to search for the information you need.
function	A section of code that, when programming, can be called by another part of the program with the purpose of returning a value.
parameter	Specifies any values that must be passed to a subroutine when it is called. Parameters are defined when a subroutine is declared.
scope	Refers to where in the program a variable or constant can be accessed.
procedure	A section of computer code that performs a specific task but does not return a value.
SQL	Structured Query Language, a language used to extract information from a database.
record	A data structure that allows for multiple variables of differing types to be held within it.
variable	A named memory address that holds a value. The value held in a variable can change as the program is running.

Structured Query Language (SQL)

- SQL is a language for storing, manipulating and retrieving data in databases.

SELECT	which fields to be returned. * can be used to indicate all fields.
FROM	which table the fields will be returned from.
WHERE	records meet a condition.
LIKE	can be used with WHERE to search for a pattern. % can be used as a wild card.

- Example:

Table : hotels					
ID	hotel_name	hotel_rating	rooms	bathroom	price
1	Water Lodge	2.3	50	En-suite	42.00
2	Fire Inn	4.2	60	Shared	42.00
3	Earthen House	4.4	251	En-Suite	39.00
4	Windy Inn	3.5	150	Shared & Ensuite	57.00
5	River View	3.8	180	En-Suite	46.00

- The SQL statement:

```
SELECT hotel_name FROM hotels WHERE rooms > 150
```

- would return

River View and Earthen House

- The SQL statement:

```
SELECT * FROM hotels WHERE hotel_name LIKE "%Inn"
```

- would return

2 Fire Inn 4.2 60 Shared 42.00

4 Windy Inn 3.5 150 Shared & Ensuite 57.00

Random Numbers

- A computer can be used to generate a random number.

```
myVariable = random(1,6)
```

Creates a random integer between 1 and 6 inclusive.

```
myVariable = random(-1.0,10.0)
```

Creates a random real number between -1.0 and 10.0 inclusive.



2D Arrays

A two-dimensional array (2D array) is a data structure that can be used to store data in a table-like format.

It is a collection of rows and columns, where each row and column is an array of elements. The elements in a 2D array can be of any type, such as integers, strings, or objects.

Table:

Name	Age	Gender
John Doe	30	Male
Jane Doe	25	Female
Peter Doe	40	Male

To store the data from the table above in a 2D array would look like:
`people = [["John Doe", 30, "Male"], ["Jane Doe", 25, "Female"], ["Peter Doe", 40, "Male"]]`

To access the array you could use could such as:

```
print(people[0][1])
```

This would print the age of John Doe, **30**

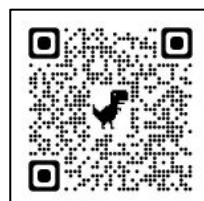
Quizizz



Extended Reading



Exam Questions



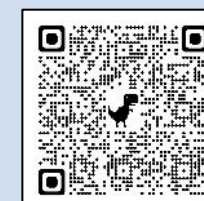
Video



Revision Techniques



Quizlet



Specification



Advanced Programming

Subprograms

- If tasks are completed more than once, these tasks can be written as functions or procedures to avoid having to type the code in again.
- Subprograms can be called whenever they are needed.
- A function is a subprogram that returns a value.
- A procedure is a subprogram that does not return a value.
- Parameters are variables used to pass values into a function or procedure.
- Structuring code into sub-programs makes the code easier to read and debug.

Defining functions and procedures

function no parameters	function with parameters
<pre>function getPi() i = 22 / 7 return pi endfunction</pre>	<pre>function sum(num1, num2) total = num1 + num2 return total endfunction</pre>
procedure no parameters	procedure with parameters
<pre>procedure greet() print("Welcome") endprocedure</pre>	<pre>procedure greet(name) print("Hello" + name) endprocedure</pre>

Calling functions and procedures

function	procedure
<pre>area = getPi() * r * r</pre>	<pre>greet("sarah");</pre>

Global and Local Variables

- Programming languages require a variable to be identified before a value is assigned to it. This is known as declaring a variable.
- The scope of a variable determines which parts of a program can access and use that variable.
- A global variable is a variable that can be used anywhere in a program. The issue with global variables is that one part of the code may inadvertently modify the value because global variables are hard to track.
- Global variables need to be defined throughout the running of the whole program. This is an inefficient use of memory resources.
- A local variable is a variable that can only be accessed within a certain block of code typically within a function. Local variables are not recognized outside a function unless they are returned. There is no way of modifying or changing the behavior of a local variable outside its scope.
- Local variables are defined only when they are needed and so have less demand on memory. Local variables only exist within the subroutine.

```
globalVar = "I am global"

def scope():
    global globalVar
    localVar = "I am local"
    print(localVar)
    globalVar = "Changed"
    print(globalVar)

scope()
```

global variable

local variable

Quizizz



Extended Reading



Exam Questions



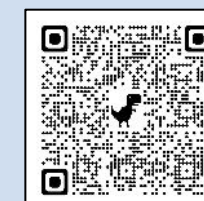
Video



Revision Techniques



Quizlet



Specification

