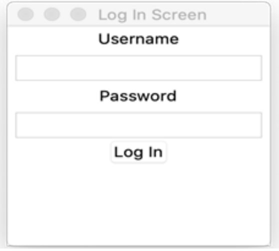
GCSE Computer Science

Component 2: Implementation 3

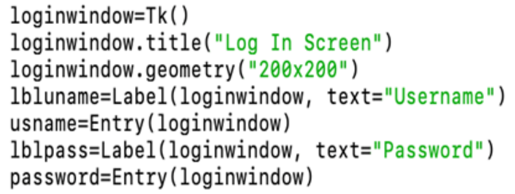
Design, write, test and refine Python 3 code

Term	Definition
Graphical user interface	A GUI is a type of interface that allows users to interact with a computer system through graphical icons.

For this qualification we will be using Python 3 and Tkinter to create the user interface.



The graphical user interface includes forms that display text boxes to enter and show information and buttons to allow the user to issue commands.



```
loginwindow=Tk()
loginwindow.title("Log In Screen")
loginwindow.geometry("200x200")
lbluname=Label(loginwindow, text="Username")
usname=Entry(loginwindow)
lblpass=Label(loginwindow, text="Password")
password=Entry(loginwindow)
```

Data Validation and verification

Term	Definition
Data Validation	Ensures that data entered is reasonable.
Data verification	Ensures that data entered is consistent.

Authentication

Authentication is the process of verifying the identity of a person or device. A common example is entering a username and password when you log in to a website or computer system.

- User enters a username
- User enters a password
- Open the file with usernames and passwords
- Loop over every username-password combination in the file
    - if the usernames match then check the passwords match, if the passwords also match then allow the user to progress

Subroutines

A subroutine in Python is known as a function and is a set of instructions designed to perform a frequently used operation within a program.

A function is a block of code that is only run when it is called. Data can be passed from the main program to the function using a method called passing parameters.
A function can return data as a result of the code being run.

```
def my_function(fname, lname):
  print(fname + " " + lname)

Start mainprogram

my_function("Fred", "Smith")
```

The program passes the values ‘Fred’ and ‘Smith’ to the function that will then print out:
Fred Smith

Self-documenting code and meaningful identifiers

The use of self-documenting code and meaningful identifiers allows third party programmers to update and amend code to improve systems. The aim is to:
Make source code easier to read and understand

- Minimise the effort required to maintain or extend legacy systems
- Reduce the need for users and developers of a system to consult secondary documentation sources such as code comments or software manuals

Annotation

Annotation is a term used in programming to refer to comments in the code to explain the purpose or role of the code.

```
def savepupil() :
    #Setting up data structure
    FirstnameSave=FirstnameVar.get()
    FirstnameSave=FirstnameSave.ljust(50)
    SurnameSave=SurnameVar.get()
    SurnameSave=SurnameSave.ljust(50)
    AddressSave=AddressVar.get()
    AddressSave=AddressSave.ljust(50)
    #Copying data from text boxes
```

Well annotated code allows a competent third party to maintain or improve a previous programmer’s code.