

National Unit Details			
Code(s)		Title(s)	
ICTPRG407		Write scripts for software applications	
ICTPRG405		Automate processes	
Assessment Task			
Number	1	Title	Generating common user accounts

Section 1 – Assessment Task Overview and Description

Student / Class Details

Full name:

Student ID:

Teacher:

Date / Time started:

Please read the whole assessment before starting, including the Summary of Evidence and Items / Criteria. Any missing files / an incomplete submission **will not be accepted, and a resit will be required.**

By checking the box below, you agree that penalties exist for plagiarized work, that all work submitted is your own and you have read the above statement. Please refer to the bottom of the document for more information on plagiarism.

☒ I confirm that all work is my own and have read the entire assessment.

Case Study and tasks start on next page.

National Unit Details			
Code(s)		Title(s)	
ICTPRG407		Write scripts for software applications	
ICTPRG405		Automate processes	
Assessment Task			
Number	1	Title	Generating common user accounts

Case Study

Dod&Gy Evil Inc. a software cracking company that has no morals and really doesn't exist has hired you for a contracting gig. They have gathered a list of employees first name, last name and age from a company called Huawow. Huawow are rumoured to be very slack with their cybersecurity. According to rumours most employees still use the autogenerated password given to them at their initial employment, as they are not required to change them. Dod&Gy have requested for you to help generate a possible email / password combination that may have been autogenerated so they can attempt to gain access to their systems.

Tasks

You must create a python script that asks for details about employees at Huawow and output a possible email password combination. The details included are their FirstName, LastName and Age respectively. Please see the below table for the rumoured format Dod&gy has collected. It also seems their age is magically offset by the 3rd last digit of **your** student ID, spooky!

Eg. If your student ID is 2534578, their age will be: 54 + 5. Please note the below table is for demonstration only, you must calculate these values dynamically depending on the input.

Input Details <u>(Not accurate)</u>	Output Example <u>(Not accurate)</u>
First Name: Mary Last Name: Bill Age: 54 (+5)	mbill@Huawow.io maryB_1961
First Name: Joe Last Name: Feathers Age: 32 (+5)	jfeathers@Huawow.io joeF_1983
First Name: Frank Last Name: Grasper Age: 66 (+5)	fgrasper@Huawow.io frankG_1949

1. Using a python script:
 - a. Read the specified details from the user (First Name, Last Name, Age).
 - i. This will mean that your application **must accept ANY name and age combination** not just the ones from the examples.
 - b. Process their age depending on your Student ID
 - i. If the 3rd last digit of your student id is a 0, do not offset the age by anything.
 - ii. You **MUST** hard code this digit in your script, do not ask for it.
 - iii. If you do not know your student ID, it is the numbers used when signing into anything Swinburne related.
 - c. Generate and output a pipe ('|') separated email and password combo.
 - i. This output will be like the example provided, but with the provided name and age used instead.
 - d. Keep asking until the user has entered **an empty first name**

National Unit Details			
Code(s)		Title(s)	
ICTPRG407		Write scripts for software applications	
ICTPRG405		Automate processes	
Assessment Task			
Number	1	Title	Generating common user accounts

More tasks on the next page.

National Unit Details			
Code(s)		Title(s)	
ICTPRG407		Write scripts for software applications	
ICTPRG405		Automate processes	
Assessment Task			
Number	1	Title	Generating common user accounts

2. Write some brief technical documentation about the python script you have created below.

E.g. What operating systems can this script run on? What version of python does it require? How much ram + disk space does it need to run?

This script can run on Windows, MacOS and Linux.

All that is required is that python interpreter is installed on the system.

There are generally four ways to run the script.

- Double clicking a python file will run the script through python interpreter.
- The script can also run from the command line in Windows, or from the terminal for MacOS and Linux.
- Thirdly, coding applications such as Atom, Sublime Text or Visual Studio code can run the python code.
- Lastly, the code itself can be converted into an executable file which allows it to be run without a python interpreter.

Python 3 is not considered to be backward compatible with python 2, so this code will only work with the python 3 interpreter.

Running the code itself draws very little system resources. Python uses under 15MB of Memory to run this code, and the size of the file created is only 1KB.

3. Create a simple user manual outlining the steps required to make this script run. (Must be in step form)

E.g. What software do I need, how can I run the program?

1. Mac computers usually have python installed, as do most Linux distributions. If, as is the case with Windows, python is not pre-installed, then visiting www.python.org provides links to install options.
2. Once python is installed, simply double-clicking a file with a .py extension will have python open and run the file.
3. Alternatively, the file can be run from within the command line or terminal.
4. Open command prompt.
5. Using the correct path, navigate to the directory containing the python file.
6. Enter the file name, inclusive of the .py extension, and press enter.
7. The code will then run from within the command line.
8. Lastly, it is possible to run and alter the code from within a coding program such as Visual Studio Code.
9. The next few steps explain how to use the program once it is running:
10. As prompted, enter the details of each employee. Starting with first name, then last name, and then finally with age.
11. Continue with this process, until the name and age details of all employees are entered.
12. When there is no more employee information to be added, simply press enter when prompted for a first name again.
13. The program will then output all automatically generated email/password combinations for each employee.

End of Assessment.

National Unit Details			
Code(s)		Title(s)	
ICTPRG407		Write scripts for software applications	
ICTPRG405		Automate processes	
Assessment Task			
Number	1	Title	Generating common user accounts

Section 2 – Assessment Task Submission Information

Submission Details	Due date:
	<ol style="list-style-type: none"> The assessment task must be submitted via Canvas or directly to the teacher with an assessment cover sheet. Ensure to include on the front page or in the header or footer of your assessment: <ul style="list-style-type: none"> your name student ID your teacher's name assessment agreement checked Submissions received after the submission date must be approved by your teacher. Incomplete submissions will not be accepted.

Summary of Evidence to be Submitted

A Python script file (.py) with code representing the task at hand

This assessment document with all form fields filled and the agreement checked

The task will be assessed as satisfactory when all the required evidence listed has been satisfactorily demonstrated.

* If applicable, for graded units, the task must be satisfactorily completed before marks will be allocated. Refer to your unit outline for more information.

National Unit Details			
Code(s)		Title(s)	
ICTPRG407		Write scripts for software applications	
ICTPRG405		Automate processes	
Assessment Task			
Number	1	Title	Generating common user accounts

Section 3 – Assessment Task Criteria and Outcome	
<i>All items/criteria must be demonstrated satisfactorily to achieve this task. The items/criteria for this activity will be assessed as S – Satisfactory or US – Unsatisfactory.</i>	
Items/criteria	
1.	Demonstrated an understanding of python string methods. At least one of: strip, split, lower, or upper
2.	Demonstrated an understanding of repeated input using a while loop and exiting that loop when required
3.	Provided basic documentation about the script
4.	Developed a basic user guide on how to use the script
5.	Demonstrated an understanding of Python Input and Output through the console.
6.	Generated an output in the correct format (as examples) using information provided from the user of the final python script.

National Unit Details			
Code(s)		Title(s)	
ICTPRG407		Write scripts for software applications	
ICTPRG405		Automate processes	
Assessment Task			
Number	1	Title	Generating common user accounts

Section 4 – General Assessment Information

Decision Making Rules	<p>Each activity in the assessment task must be satisfactorily completed for the task to be assessed as satisfactory.</p> <p>Every task must be satisfactorily completed to be assessed as competent in the unit.</p> <p><i>* For graded units, competence must be demonstrated before a mark can be given.</i></p>
Plagiarism	<p>There are serious penalties for plagiarism that may include repeating a new assessment task or being withdrawn for the unit / course.</p> <p>Students must ensure that all assessments are their own work (or group work and clearly noted as such).</p> <p>Please refer to www.swinburne.edu.au/corporate/registrar/plagiarism/index.html</p>
Reasonable Adjustment	<p>Students may request reasonable adjustment for assessment tasks.</p> <p>Reasonable adjustment usually involves varying:</p> <ul style="list-style-type: none"> the processes for conducting the assessment (eg: allowing additional time, varying the venue) the evidence gathering techniques (eg: oral rather than written questioning, use of a scribe, modifications to equipment) <p>However, the evidence collected must allow the student to demonstrate all requirements of the unit.</p> <p>If you have any other issue that may impact your ability to undertake the assessment, please discuss with your teacher.</p>
Re-submission (where tasks are not satisfactorily completed)	<p>Assessment tasks that are not satisfactory can be resubmitted up until the end of the unit as scheduled on the Unit Outline. The timing on this may depend on the equipment required for this assessment task.</p> <p>Resubmissions received after the scheduled unit end date may not be accepted unless approved by the teacher prior to the end date.</p> <p>Note: Assessment tasks submitted for the first time after the unit end date as scheduled in the Unit Outline will not be assessed and the student should re-enrol into the unit.</p>
Special consideration	<p>Students may apply for Special Consideration where personal circumstances have adversely affected their task result or ability to undertake an assessment. A Special Consideration form can be completed prior to, but no later than 3 days after, the date of assessment and submitted to the relevant Manager.</p>
Work Health & Safety	<p>Activities may require the use of equipment or participation in group exercises. If the teacher identifies any unsafe activity or potentially dangerous situations, the teacher can stop the assessment at any time.</p>