|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Task:** | | **AT1** | | |
| **Task Title:** | | **Portfolio** | | |
| **Task Code:** | | **AT1 POR Part 1** | | |
| **Name:** | | **Ashby Scattini** | | |
| **Student ID:** | | **20146307** | | |
|  | |  | | |
| Assessment type (): | | | | |
|  | Questioning (Oral/Written) | |  | Portfolio |
|  | Practical Demonstration | |  | Project |
|  | 3rd Party Report | |  | Other – Please Specify |
|  |  | |  |  |
| Assessment Resources | | | | |

|  |
| --- |
| The base requirements this assessment task include:   * Python interpreter * IDE or editor for developing Python programs (only PyCharm supported by the college) * Git tools (GitHub tools) for version control * Access to the internet * Access to Office 365 or Microsoft Word * Access to Blackboard   Use of some of these items may not occur in this part of the assessment task. |
| Assessment Due This assessment is split into components that have several due dates:   |  |  |  | | --- | --- | --- | | **Task** | **Week** | **Date** | | Task 1 | 6 | 17:00, Friday of week |   Refer to Blackboard for most accurate dates, which may alter due to unforeseen circumstances.  We also will endeavour to update these document(s) at the same time. |
| Instructions Follow the steps listed in this assessment item.  Submission of the documentation, code, and associated items is at the end of each part of the portfolio.  Each part of the portfolio has a deadline for submission.  It is advantageous to you to attempt to meet the deadline provided. |
| Important If you are using a different configuration of tools and equipment for this assessment item, then assistance in this and subsequent parts of the portfolio to ensure the systems work correctly will be limited.  Make sure you put your name and student ID on the front page. Failure to do so will mean your submission will not be marked. |
| Scenario You have just started working for a boutique Perth-based software development company. The company uses git for source control and GitHub for hosting their git repositories and providing remote access.  To foster a culture of collaboration, the company also requires you to set up a GitHub pages website to introduce yourself and blog about your progress.  GitHub pages is a free service that renders a specially-named git repository as a website. The repository can contain markdown or HTML.  Your workstation will already have the necessary tools (git) installed. However, it is not yet configured for the workplace. Moreover, the company has an open policy on BYO devices so you may want to install Git on your own computer and be mindful of settings that need to be applied globally. |
| Answering Questions When a step includes a question, you must attempt to answer it.  There is a minimum and maximum number of words to use for each answer.  If a step has more than one question, these maxima and minima are a total for all the questions in that specific step.  All answers must be in complete sentences unless indicated.  If required, make sure to add any code you’ve written in a separate file to your submission. DO NOT put code in a Word document. |
| Sources of Information In industry, it is good practice to keep track of where information was obtained. This is especially true if it is a written document, or even code.  If you answer any questions using information from web sites, please include the site name and URL (Web site address) after the answer. Likewise, include the title and author for books and magazine articles. For example:   * RS Electronics Ltd: <https://au.rs-online.com/> * Slack API Documentation, Users List Method: <https://api.slack.com/methods/users.list> |
| Code Storage We advise that you create a GIT repository on GitHub and use this to store a copy of your work.  You may also use OneDrive within your college Office365 to store a backup of your code or keep a copy on a USB thumb drive.  Please note that it is the student’s responsibility to keep backups of their work. This includes any documents and code. |

|  |  |  |
| --- | --- | --- |
| **STEP** | **Task to perform** | Words Min/Max |
| 00 | Example question This is an example step. It may contain instructions or a question.  If a minimum and maximum word count are provided, you must take those into account. |  |
|  | Your answer or evidence (e.g., screenshots) go in this box.  This box will grow automatically as you put in text and images. |  |
| 01 | Create GitHub account Your company’s policy is to use GitHub (and git) for version control. Any new repositories will be created in GitHub and cloned locally.  If you haven’t already done so, please create a new GitHub account. You may use your student email address to set up the account or use your own email address.  Follow the instructions provided by GitHub and/or your lecturer to set up your account and provide evidence (screenshots) below.  If you already set up a GitHub account earlier, also provide evidence (screenshots) below. | n/a |
|  |  |  |
| 02 | Local Installation The company communicates with you that to install and configure git locally, you will need to ensure the following:   * The default branch name should be set to main * You **must** configure Git with your employee number [use student number] as the username and company email [use your TAFE email]  1. Research and write down the git commands needed to do the above and then run them locally. Include a screenshot to evidence your actions. 2. Do these settings apply to all repositories? Qualify your answer. 3. What disruptions to collaborative work do you think could occur if company personnel were not given these specific instructions? 4. Why does the organisation need to give you these instructions as opposed to centrally configure them? | 3-5 sentences |
|  | *B* – Yes, these settings apply **globally** to all repositories created or cloned on your system. The –global sets these for your entire Git environment.  C – If personnel were not instructed to follow these settings, several issues could pop up such as mismatched author information.  D – The organisation provides these instructions because Git settings are managed locally making centralised control hard, having procedures ensures uniformity. |  |
| 03 | Create personal repository In your account, create a new repository following the procedures from GitHub or provided by your lecturer.  In this case, make sure you name the repository username.github.io. and replace username with your own GitHub username.  You **must** follow this convention, or it will not work properly.  Provide evidence (screenshots) of your newly created repository down below. | n/a |
|  |  |  |
| 04 | Create a local clone In this step, you will create a local clone of the repository you created in the previous step. It is assumed you will be running all git commands in a terminal (sometimes called console, or command line).  Use the instructions from GitHub or provided by your lecturer to create a local clone of the new repository on your workstation.  Run git remote -v to show the local repository’s remote counterpart.  Provide evidence of the clone and the directory into which it was cloned below (use screenshots). | n/a |
|  |  |  |
| 05 | Create a personal web page There are multiple steps to this question. Document **each step** with a screenshot.   1. In your local directory, create a file called index.md*[[1]](#footnote-2)*. Add the following line to this, where you replace Name with your own name:  # Name 2. Add the file to your “staging” environment in the main branch (the default branch):  git add index.md 3. Commit the file with a suitable commit message:  git commit -m ‘First commit’ 4. Run the following command to investigate the local repository’s status:  git status 5. Push your changes to the remote repository:  git push 6. Check that your page is now live at username.github.io, where username is, of course, your own username. | n/a |
|  |  |  |
| 06 | Making a change Your personal web page on GitHub Pages is still a bit, well, bare. So, you decide to add some information to it. You’re not too worried about making it pretty, yet.  Follow the instructions and **provide evidence** **for each step**.  Create a local branch:  git checkout -b update-profile  Check that you are indeed working on your new branch:  git status  Open the file index.md in your favourite editor and add the following lines after the existing markdown code (please, customise them for your situation, especially the date):  ### Profile - \*\*Started\*\*: **today’s date** - \*\*Role\*\*: Software Developer - \*\*Hobbies\*\*: Rowing, Reading, and Role-playing  Add index.md to the staging and commit it:  git add index.md git commit -m “Updated my profile”  Push your changes to the remote repository. The command looks a bit different because you are working on a new branch:  git push --set-upstream origin update-profile  **Important!** Take notice of the information that this command has provided. You will need this in the next step. | n/a |
|  |  |  |
| 07 | Creating a Pull Request To integrate your changes into your actual profile page, you will need to merge the changes into the main branch. But first, you will create a Pull Request (PR).  The previous command provided a link with a message that said Create a pull request (…) by visiting:  Open this link in your web browser (you should be logged in to GitHub. If you are not logged in: do so first.)  You can now create a so-called Pull Request. GitHub will pre-fill most fields, so you can simply click “Create pull request”.  A screenshot of a computer  Description automatically generated with medium confidence  GitHub will create a new Pull Request. It will have a name and a number, for example, Updated profile #1.  Provide evidence of your Pull Request below (screenshot). | n/a |
|  |  |  |
| 08 | Merging the Pull Request The final part to this process is to merge the Pull Request (remotely and locally). GitHub provides the tools needed to do that.  Make sure you have the page with the Pull Request open in GitHub. It should look like this:  Graphical user interface, text, application, website  Description automatically generated  There are different ways of merging, but in this instance, you may use the default ‘Merge pull request’ as suggested by GitHub.  Click on ‘Merge pull request’ (this will add a new text input on the screen, which you may ignore), then confirm the merge by clicking ‘Confirm merge’.  **Important: do not delete the Pull request at this stage.**  Reload your personal page in a browser. It may take a few minutes for the changes to show up. (Reminder: the link is username.github.io.)  Update your local repository:   1. First, switch back to your main branch:  git checkout main 2. Next, fetch the changes from your remote repository:  git pull   This will ensure your local repository is again synchronised with your remote repository.  Provide evidence of the entire merging process in the form of screenshots. | n/a |
|  |  |  |
| 09 | Inform required personnel of the work The last step of this assignment is to inform required personnel of the work you have done.  In this case, you will inform your lecturer**. But ensure you inform them as though they are your line manager.**  Write them a polite email saying you have done the work. Include the following information:   * Your repository on GitHub (link) * The link to your personal profile page   You do not need to send the email but include a screenshot below | n/a |
|  |  |  |
|  | Submission of Portfolio Work To submit the portfolio, do the following:   * Ensure you have put your name and student ID on the front page of this document. Your submission will not be accepted if name or student ID is missing. * Save the document with your answers as a MS Word file (.docx). * Do NOT zip your Word document! * Open Blackboard, and locate the AT2 Portfolio Task 1 assessment * Open the assessment and upload the original document. * Click submit.   Whilst there is no need to use any other word processing software as you have access to Office 365 for free as a student, if you use Apple Pages, or Open Office, we will then require you to upload the original file **AND** a PDF version. |  |

# Appendix A: Code Style Guidelines

Your code will follow the PEP 8 standard.

Readability Counts  
- Zen of Python

Explicit is better than implicit.  
- Zen of Python

Other code standards available in the Presentation, “Python Coding Standards for North Metropolitan TAFE”.

1. You can use either markdown (index.md) or html (index.html) [↑](#footnote-ref-2)