|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **PYTHON/MySQL PROJECT CHALLENGE RUBRIC** | | | | | | | | |
|  | **Database Principles using** | | | | | | **Possible Mark** | **Mark Obtained** |
|  | **Understanding problem as reflected by the design structures** | | | | | |  |  |
| **Design specifications** | 8 | 6 | | | 4 | 2 | **8** |  |
| Used more than one appropriate design structure. e.g. Flowcharts and Algorithm depicting an understanding of problem | Design structures implemented but with logical errors | | | There is an indication of use of design structures | An attempt made but the designs are wrong |  |  |
|  | **Appropriate variable naming conventions** | | | | | |  |  |
| 4 | 3 | | | 2 | 1 | **4** |  |
| Variable naming follow specific convention and its uniform | Variable naming is consistent but with a few not meaningful | | | Half of the variables are not meaningful | No conventions used. Variable names not meaningful |  |  |
| **Use of Selection Structures (IF, ELIF, Else )** | | | | | | | |
| 8 | 6 | | | 4 | 2 | **8** |  |
| Used appropriate and most effective selection structures to solve the problem in all instances | Used appropriate and most effective selection structures in most instances | | | Appropriate and most effective use of selection structures in less than 50% of the instances | Inappropriate or ineffective use of selection structures |  |  |
| **Use of iteration structures. (For loop, Do While, )** | | | | | |  |  |
| 8 | 6 | | | 4 | 2 | **8** |  |
| Used appropriate and most effective repetition structures to solve the problem in all instances | Used appropriate and most effective repetition structures in most instances | | | Appropriate and most effective use of repetition structures in less than 50% of the instances | Inappropriate or ineffective use of repetition structures |  |  |
| **Program Solution** | **Use of list or stacks or arrays or dictionaries or queues.** | | | | | |  |  |
| 8 | 6 | | | 4 | 2 | **8** |  |
| Used more than one structure correctly | Implemented only one structure correctly | | | Attempt to use structure done but with some errors | Limited to no attempt of using any of the above collections |  |  |
| **Use of Functions, (parameter passing) including in-built functions** | | | | | |  |  |
| 8 | 6 | | | 4 | 2 | **8** |  |
| Excellent interaction/ communication between modules/classes. Includes parameter passing | Proficient/appropriate use of modules/functions or other sub-procedures with small flaws. Includes parameter passing between modules/ functions or other sub-procedures but not always appropriate or correct. | | | Limited use of modules/functions or other sub-procedures. Limited parameter passing | No use of modules/functions or other sub-procedures. No parameters passed |  |  |
| **File handling** | | | | | |  |  |
| 4 | 3 | | | 2 | 1 | **4** |  |
| Excellent uses of text files and works perfectly | Text files implemented but few errors | | | Attempt to use text files but with errors | text file defined but not used |  |  |
| **Software testing- doctest and unittest** | | | | | |  |  |
| Any of docstring/unittest implemented | | | Attempt to test the code but with errors | | Only import unittest/ doctest used but not test done |  |  |
| 4 | | 2 | | | 1 | **4** |  |
| **Complexity of Code** | | | | | |  |  |
| 8 | 6 | | | 4 | 2 | **8** |  |
| Advanced. Other techniques such as GUI implemented | Skilled | | | Intermediate | Basic |  |  |
|  | **Error/Exception Handling** | | | | | |  |  |
| 8 | 6 | | | 4 | 2 | **8** |  |
| Excellent exception handling by the use of selection structures and try, catch statements. Proper Feedback given to user. | Exceptions minimized by the use of selection structures and try, catch statements. No Feedback given to user | | | Exception minimized by the use of selection structures only. No Feedback given to the user. | Attempt to implement exception resulted in logic or syntax errors |  |  |
| **Program Compiles Successfully** | | | | | |  |  |
| 8 | 6 | | | 4 | 2 | **8** |  |
| No run time or syntax errors. All the options are executed successfully | Program produce runtime errors only | | | Some of the options produce errors when executed | Majority of the code produce errors. No evidence of tests |  |  |
| **Output** | | | | | |  |  |
| 8 | 6 | | | 4 | 2 | **8** |  |
| Output correct Matches the user requirements | Output is correct but with not meeting user requirements in all path | | | Some sections produce incorrect output | Incorrect logic. Not meeting user requirements |  |  |
| **Program Documentation** | | | | | |  |  |
| 8 | 6 | | | 4 | 2 | **8** |  |
| Code well documented in all paths | 75% of code documented | | | Only 50%of code is documented | Just 20% of the code is documented |  |  |
| **General Evaluation** | **Time Management: (0 = always late, work was never done)** | | | | | |  |  |
| 8 | | | 6 | | 4 | **8** |  |
| All deadlines met | | Project submitted few hours after due date | | | Project submitted after interventions |  |  |
|  | | | | | | **100** |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **FINAL COMPETENCE LEVEL INDICATORS** | | | | | |
| **Outstanding** | **Highly competent** | | **Competent** | **Not yet competent** | **Not achieved** |
| **5** | **4** | | **3** | **2** | **1** |
| **Student’s Competence Level:** | | **PASS ☐ FAIL ☐** | | | |
| **Student’s Name and Signature:** | |  | | | |
| **Lecturer’s Signature:** | |  | | | |
| **Date:** | |  | | | |
| **Feedback to student-(Facilitators are expected to give students constructive feedback on positive areas and also areas which require more attention)** | | | | | |