|  |  |  |  |
| --- | --- | --- | --- |
| **Criteria for Evaluation** | | **Points Total: 50** | |
| **Code** | | **Subtotal: 40** | |
| A user can choose a location by name (not station ID number) for tide predictions. | | 5 | |
| The user can choose from a list of at least 3 tide stations (there are 12 harmonic stations in Oregon). | | 3 | |
| A user can choose a date for the tide prediction. | | 5 | |
| If data for the location and date selected by the user are in the database, that information is used | | 5 | |
| If data for the user’s selection isn’t in the database, it is downloaded from the web service and displayed | | 12 | |
| When data is downloaded from the web service, it is stored in the database without inserting duplicate information. | | 6 | |
| The app doesn’t crash if the user makes a selection when there is no Internet connection. If no data is available a suitable message is displayed. | | 4 | |

Comments:

|  |  |
| --- | --- |
| **Code Quality** | **Subtotal: 10** |
| Does the program run without crashing? | 3 |
| Is there good “separation of concerns”, for example, is parsing handled in a separate class? | 2 |
| Is proper indentation used? | 5  (one point deducted for each “no” answer) |
| Are the UI elements named according to convention (example: incrementButton)? |
| Are Java variables, constants, methods, and classes named using descriptive names and correct casing? |
| Have any unnecessary lines of code or files been removed? |
| Are there explanatory comments in the code? |
| Are all other good object oriented programming practices followed? |

Comments:

|  |  |
| --- | --- |
| **Total** |  |