# Quiz App Scores Project Short Handout

| Getting Started  Open App Inventor with your finished Quiz App from the last lesson.  Modify your app to keep score of how many questions are answered correctly and incorrectly. Be sure and restrict it so that the quiz taker can only receive credit for answering each question once (i.e., if there are three questions, the quiz taker can only be credited with three correct answers) |  |
| --- | --- |

# Designing the User Interface (UI)

| **UI Component** | **Name** | **Properties** |
| --- | --- | --- |
| Label | ScoreLabel | Put somewhere on the screen.  Type in “Score: 0/0” |

# Coding the App: Version 1 Simple Scoring

| **Data Abstraction** | | **Values** |
| --- | --- | --- |
| Score Variable | | Set to 0 |

| **Event Handlers** | | **Algorithms** |
| --- | --- | --- |
| AnswerButton.Click | | Modify so that if the user gets the answer correct, increment the score (add 1 to it) and display it. |

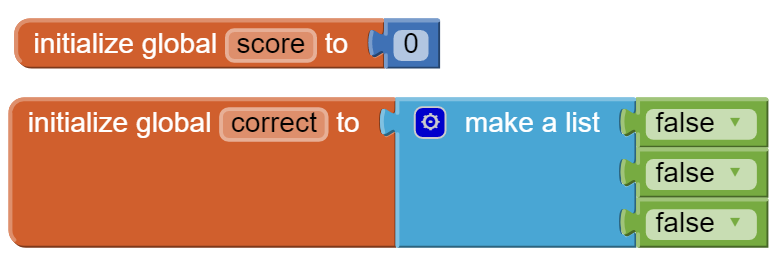
# Testing the App

| **Inputs** | | **Expected Outputs** | **Actual Outputs** |
| --- | --- | --- | --- |
| Type in correct answer | | Score goes up | ? |
| Type in the correct answer on the same question | | Score goes up again! (we will change this in the next version) |  |
| Type in wrong answer | | Score does not change | ? |

# 

# Coding the App: Version 2 No Double Scoring

| **Data Abstraction** | | **Values** |
| --- | --- | --- |
| Correct List | | Set to a list of falses (for each question). This will keep track of whether they have answered that question correctly yet. |



| **Event Handlers** | | **Algorithms** |
| --- | --- | --- |
| AnswerButton.Click | | Modify so that if the user gets the answer correct and if the correct list is still false at that index, increment the score (add 1 to it) and display it and change the correct list to true at that index using the replace list item block.  If the user gets the answer wrong but had answered correctly last time according to the correct list, subtract a point from score, display, and change the correct list to false. |

# Hints

When the answerButton is clicked, inside the if block if they get the answer correct, make sure they haven’t gotten credit for that question yet by adding another *if* block to see if the correct list at that index is still false (use the *select item from list* block with the index variable). If it is still false, add 1 to a score variable and display it. Then change the correct list at that index to true using the *replace item in list* block.

Then, look at the else part of the answerButton.Click. If they answered incorrectly and they had previously answered correctly, you must subtract 1 from their score and set correct item now to false. You will need to use another if block in here as well mirroring what you did in the last step.

# Testing the App

| **Inputs** | | **Expected Outputs** | **Actual Outputs** |
| --- | --- | --- | --- |
| Type in correct answer | | Score goes up | ? |
| Type in the correct answer on the same question | | Score does not change! | ? |
| Type in wrong answer on the same question | | Score goes down. | ? |
| Type in the wrong answer on another question that has not been answered yet. | | Score does not change. | ? |

# Iterative Design: Enhancements

Some possible enhancements are to give the correct answer if they get it wrong, to print out encouraging feedback, to use TextToSpeech, to use background colors to indicate right or wrong, to give hints, etc.