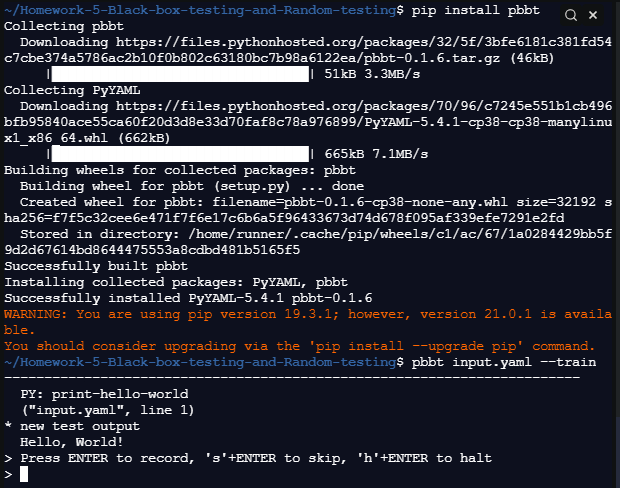
<https://github.com/AJ-Protzel/HW5-Black-Box>

1. In test\_functions.py, 2 classes are used (ValidQ1, InvalidQ1), with two functions in each for test cases for equivalence partitioning and BVA boundaries.
   1. Equivalent partition: -15(invalid), 15(valid), 60(invalid)
      1. Condition: less than 1, more than 31
   2. BVA: 0(invalid), 1(valid), 31(valid), 32(invalid)
      1. Below: 0, 31
      2. Above: 1, 32
2. In test\_functions.py, 2 classes are used (ValidQ2, InvalidQ2), with two functions in each for test cases for equivalence partitioning and BVA boundaries.
   1. Equivalent partition: ""(invalid), "aaaaaaaa"(valid), "aaaaaaaaaaaaaaa"(invalid)
      1. Condition: less than 7, more than 10
   2. BVA: "aaaaaa"(invalid), "aaaaaaa"(valid), "aaaaaaaaaa"(valid), "aaaaaaaaaaa"(invalid)
      1. Below: 6, 10
      2. Above: 7, 11
3. In test\_functions.py, 2 classes are used (ValidQ3, InvalidQ3), with two functions in each for test cases for equivalence partitioning and BVA boundaries.
   1. Equivalent partition: 10(invalid), 30(valid), 63(invalid), 67(valid), 90(invalid)
      1. Condition: less than 16 and 65, more than 70 and 60
   2. BVA: 15(invalid), 16(valid), 59(valid), 60(invalid), 65(invalid), 66(valid), 70(valid), 71(invalid)
      1. Below: 15, 59, 65, 70
      2. Above: 16, 60, 66, 71
4. The installation and execution is very simple. I had to add parentheses around the “Hello World” in the file but that was it. (the screenshot show the first step but more was done on the program on github)



1. Unit testing is black box if you write the tests before you make the function, meaning you do not know how the function will behave when given data. White box would be if you make the tests during or after the creation of the function, meaning you know how the function works and you have a testing bias.

(Twitter type social media)

* 1. Black box:
     1. Type in many characters into my post to test character limits.
     2. Try attaching multiple images to a single post.
  2. Unit testing:
     1. Test that the post accepts a string of length 0-255