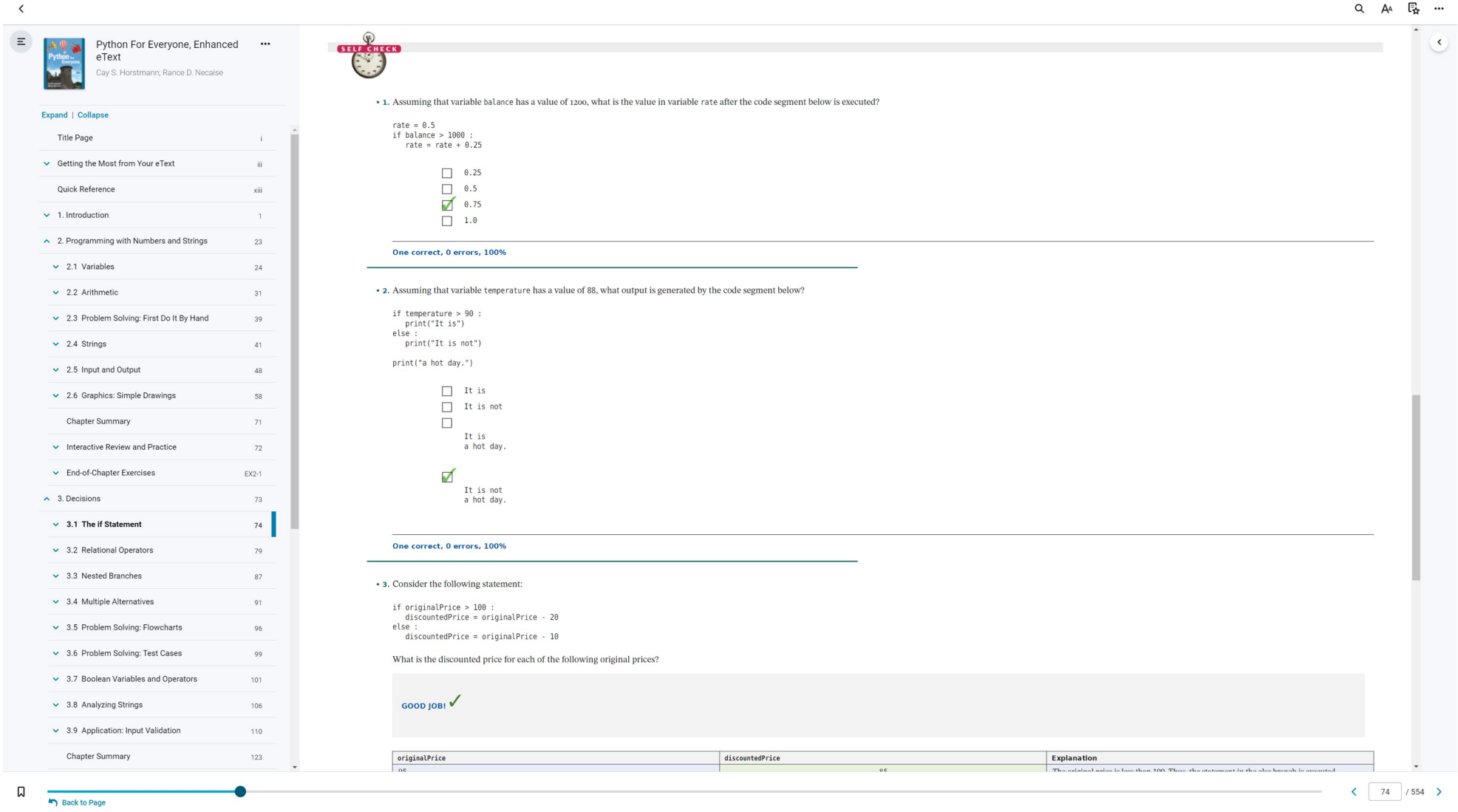
Albert Nguyen
CS131-47853 MW 06:00 PM – 08:30 PM
Chapter 3 Homework – 3.1 – 3.9



• 3. Consider the following statement:

```
if originalPrice > 100 :
   discountedPrice = originalPrice - 20
else :
   discountedPrice = originalPrice - 10
```

What is the discounted price for each of the following original prices?

GOOD JOB! ✓

originalPrice	discountedPrice	Explanation
95	85	The original price is less than 100. Thus, the statement in the else branch is executed.
100	90	The original price is not greater than 100. Again, the statement in the else branch is executed.
105	85	Because the original price is greater than 100, the statement in the first branch is executed.

3 correct, 0 errors, 100%, 22 seconds

Start over

• 4. Consider the following statement:

```
if hour < 21 :
   response = "Goodbye"
else :
   response = "Goodnight"
```

Determine the value of response when hour has the values given in the table below. (You do not need to put quotes around the response strings.)

GOOD JOB! ✓

hour	response	Explanation
20	Goodbye	20 < 21, and the first branch of the statement executes.
22	Goodnight	It is not true that 22 < 21, so the else clause executes.
21	Goodnight	It is also not true that 21 < 21.
3	Goodbye	Because 3 < 21, the first branch executes, even though one might want to wish a good night at 3 am.

4 correct, 0 errors, 100%, 21 seconds

Start over

•• 5. For this activity, trace through the code by clicking on the line that will be executed next.

Observe the inputs as they appear in the table below. The variables fuelCapacity and fuelAmount hold the size of a vehicle's fuel tank and the actual amount of fuel in the tank. If less than 20 percent is remaining in the tank, a status light should show a red color; otherwise it shows a green color.

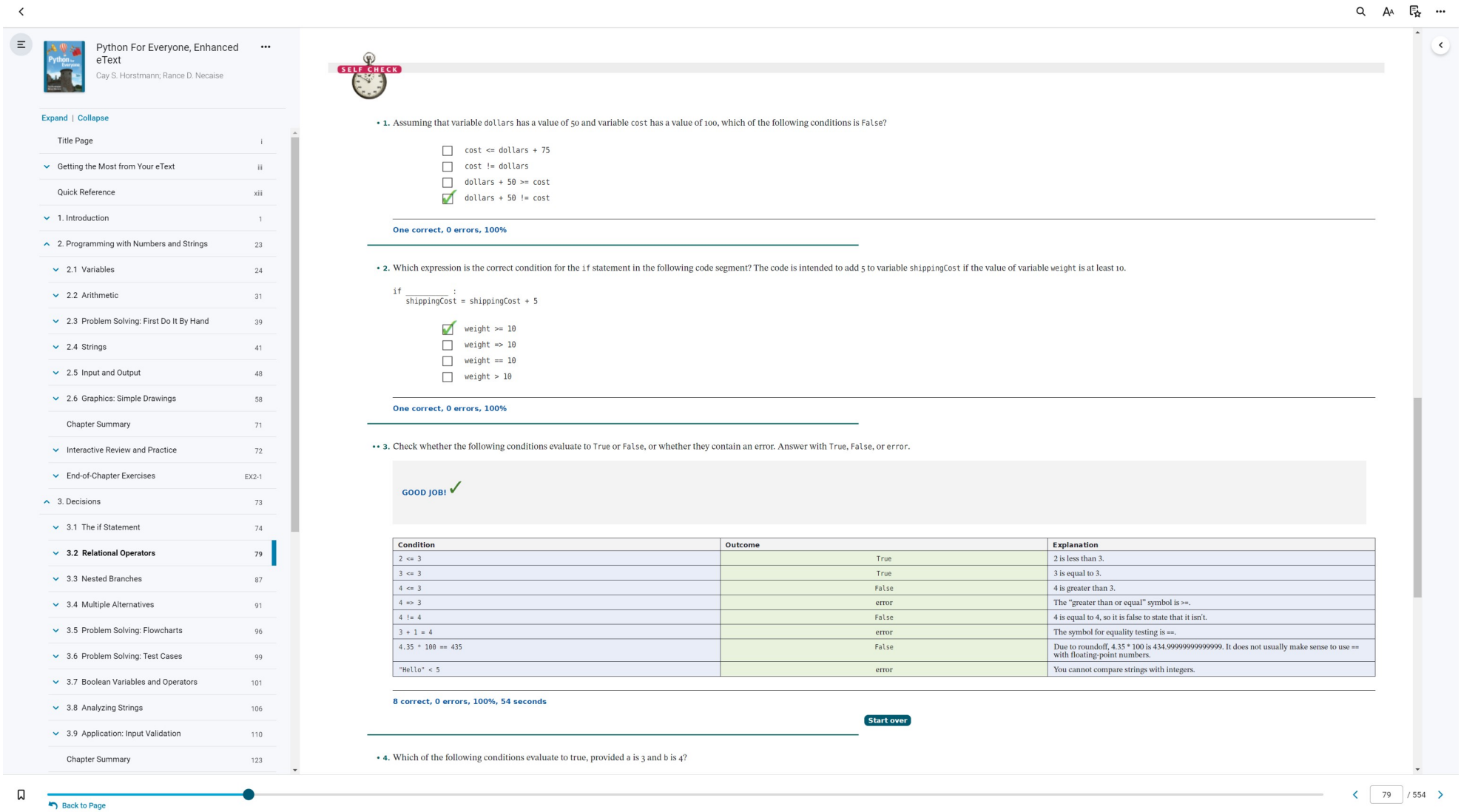
GOOD JOB! ✓

```
userInput = input("Amount of fuel: ")
fuelAmount = float(userInput)
userInput = input("Tank capacity: ")
fuelCapacity = float(userInput)
if fuelAmount < 0.20 * fuelCapacity :
   print("red")
else :
   print("green")
```

userInput	fuelAmount	fuelCapacity	Output
"10"	10		
"14"		14	green
"2.5"	2.5		
"14"		14	red
"4"	4		
"18"		18	green

3 correct, 0 errors, 100%, 113 seconds

Start over


Python For Everyone, Enhanced eText
Cay S. Horstmann; Rance D. Necaise
Expand | Collapse
Title Page i
Getting the Most from Your eText iii
Quick Reference xiii
1. Introduction 1
2. Programming with Numbers and Strings 23
2.1 Variables 24
2.2 Arithmetic 31
2.3 Problem Solving: First Do It By Hand 39
2.4 Strings 41
2.5 Input and Output 48
2.6 Graphics: Simple Drawings 58
Chapter Summary 71
Interactive Review and Practice 72
End-of-Chapter Exercises EX2-1
3. Decisions 73
3.1 The if Statement 74
3.2 Relational Operators 79
3.3 Nested Branches 87
3.4 Multiple Alternatives 91
3.5 Problem Solving: Flowcharts 96
3.6 Problem Solving: Test Cases 99
3.7 Boolean Variables and Operators 101
3.8 Analyzing Strings 106
3.9 Application: Input Validation 110
Chapter Summary 123
SELF CHECK
• 1. Assuming that variable dollars has a value of 50 and variable cost has a value of 100, which of the following conditions is False?
cost <= dollars + 75
cost != dollars
dollars + 50 >= cost
dollars + 50 != cost
One correct, 0 errors, 100%
• 2. Which expression is the correct condition for the if statement in the following code segment? The code is intended to add 5 to variable shippingCost if the value of variable weight is at least 10.
if _____ :
shippingCost = shippingCost + 5
weight >= 10
weight => 10
weight == 10
weight > 10
One correct, 0 errors, 100%
•• 3. Check whether the following conditions evaluate to True or False, or whether they contain an error. Answer with True, False, or error.
GOOD JOB!
Condition | Outcome | Explanation
2 <= 3 | True | 2 is less than 3.
3 <= 3 | True | 3 is equal to 3.
4 <= 3 | False | 4 is greater than 3.
4 => 3 | error | The "greater than or equal" symbol is >=.
4 != 4 | False | 4 is equal to 4, so it is false to state that it isn't.
3 + 1 = 4 | error | The symbol for equality testing is ==.
4.35 * 100 == 435 | False | Due to roundoff, 4.35 * 100 is 434.99999999999999. It does not usually make sense to use == with floating-point numbers.
"Hello" < 5 | error | You cannot compare strings with integers.
8 correct, 0 errors, 100%, 54 seconds
Start over
• 4. Which of the following conditions evaluate to true, provided a is 3 and b is 4?
Back to Page
79 / 554

3.2 (p.1)

Albert Nguyen
CS131-47853 MW 06:00 PM – 08:30 PM
Chapter 3 Homework – 3.1 – 3.9

		with floating-point numbers.
`"Hello" < 5`	error	You cannot compare strings with integers.

8 correct, 0 errors, 100%, 54 seconds

Start over

• **4.** Which of the following conditions evaluate to true, provided `a` is 3 and `b` is 4?

✓ True False `a + 1 <= b`

Because `a + 1` is 4, the condition reduces to `4 <= 4`, which is true.

✓ True False `a > b - 1`

Because `b - 1` is 3, the condition reduces to `3 > 3`, which is not true. The two sides are equal.

✓ True False `a >= b - 1`

Both sides of the relational operator have a value of 3. Thus, they are equal.

✓ True False `a * 2 != b + 2`

Both `a * 2` and `b + 2` are equal to 6. Thus both sides of the relational operator are equal.

4 correct, 0 errors, 100%

•• **5.** Provide conditions for the given tasks.

GOOD JOB! ✓

Task	Condition	Explanation
Give the opposite of the condition `floor > 13`	`floor <= 13`	Note that the opposite of > is <= and not <.
A programmer wrote the condition `floor = 13` to test whether the floor is 13. Fix the error.	`floor == 13`	You need to use ==, not =, to compare whether two values are equal.
Supply a condition in this `if` statement to test whether the user entered a Y: `response = input("Enter Y to quit.")` `if . . . :` `  print("Goodbye")`	`response == "Y"`	You can compare strings with ==.
How do you test that a string `response` is the empty string?	`response == ""`	You can use `response == ""` or `len(response) == 0`.

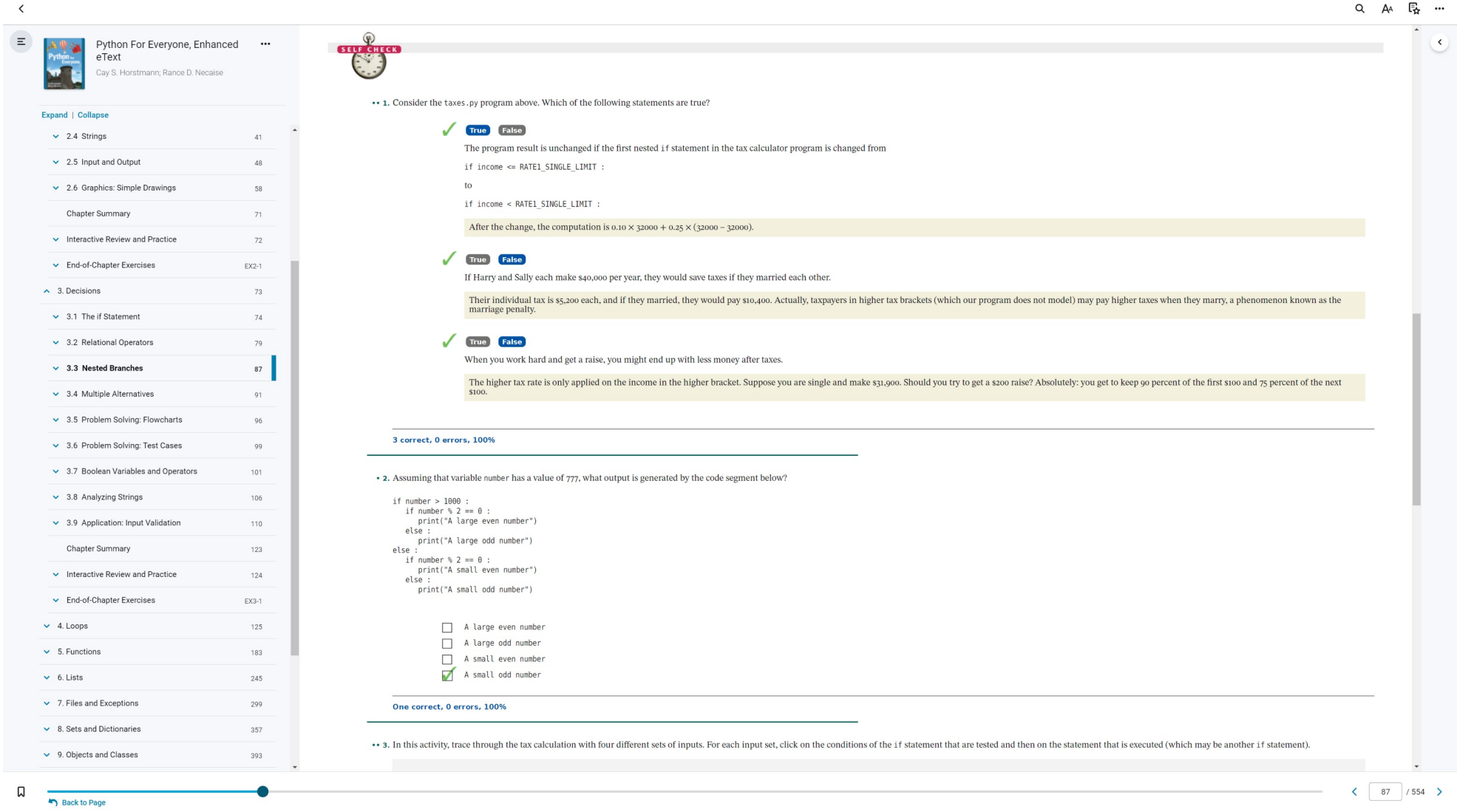
4 correct, 0 errors, 100%, 52 seconds

Start over

Back to Page

3.2 (p.2)

Albert Nguyen
CS131-47853 MW 06:00 PM – 08:30 PM
Chapter 3 Homework – 3.1 – 3.9



3.3 (p.1)

•• 3. In this activity, trace through the tax calculation with four different sets of inputs. For each input set, click on the conditions of the `if` statement that are tested and then on the statement that is executed (which may be another `if` statement).

GOOD JOB! ✓

```
income = float(input("Income: "))
maritalStatus = input("Marital status: ")
if maritalStatus == "s" :
   if income <= 32000 :
      tax = 0.10 * income
   else :
      tax = 3200 + 0.25 * (income - 32000)
else :
   if income <= 64000 :
      tax = 0.10 * income
   else :
      tax = 6400 + 0.25 * (income - 64000)
```

maritalStatus	income	tax
	~~30000~~	
~~s~~		~~3000~~
	~~40000~~	
~~s~~		~~5200~~
	~~40000~~	
~~m~~		~~4000~~
	70000	
m		7900

12 correct, 0 errors, 100%, 98 seconds

Start over

•• 4. Consider the code segment below:

```
if average >= 60 :
   print("Passing")
   if average < 70 :
      print("but just barely")
else :
   print("Failing")
```

and answer each of the following questions.

GOOD JOB! ✓

Question	Answer	Explanation
If the variable `average` contains a value of 45.0, what output is generated?	Failing	The first condition (`average >= 60`) is false, which results in the execution of the statement in the else branch.
If the variable `average` contains a value of 87.4, what output is generated?	Passing	The first condition (`average >= 60`) is true, but the second (`average < 70`) is not.
What is the smallest value for `average` that will result in both `Passing` and `but just barely` being printed?	60.0	`Passing but just barely` is printed for values greater than or equal to 60 and less than 70.
What is the largest value for `average` (to one decimal place) that will result in `Failing` being printed?	59.9	Floating-point values are not rounded before they are compared. Thus, 59.9 is the largest value to one decimal place that is less than 60.

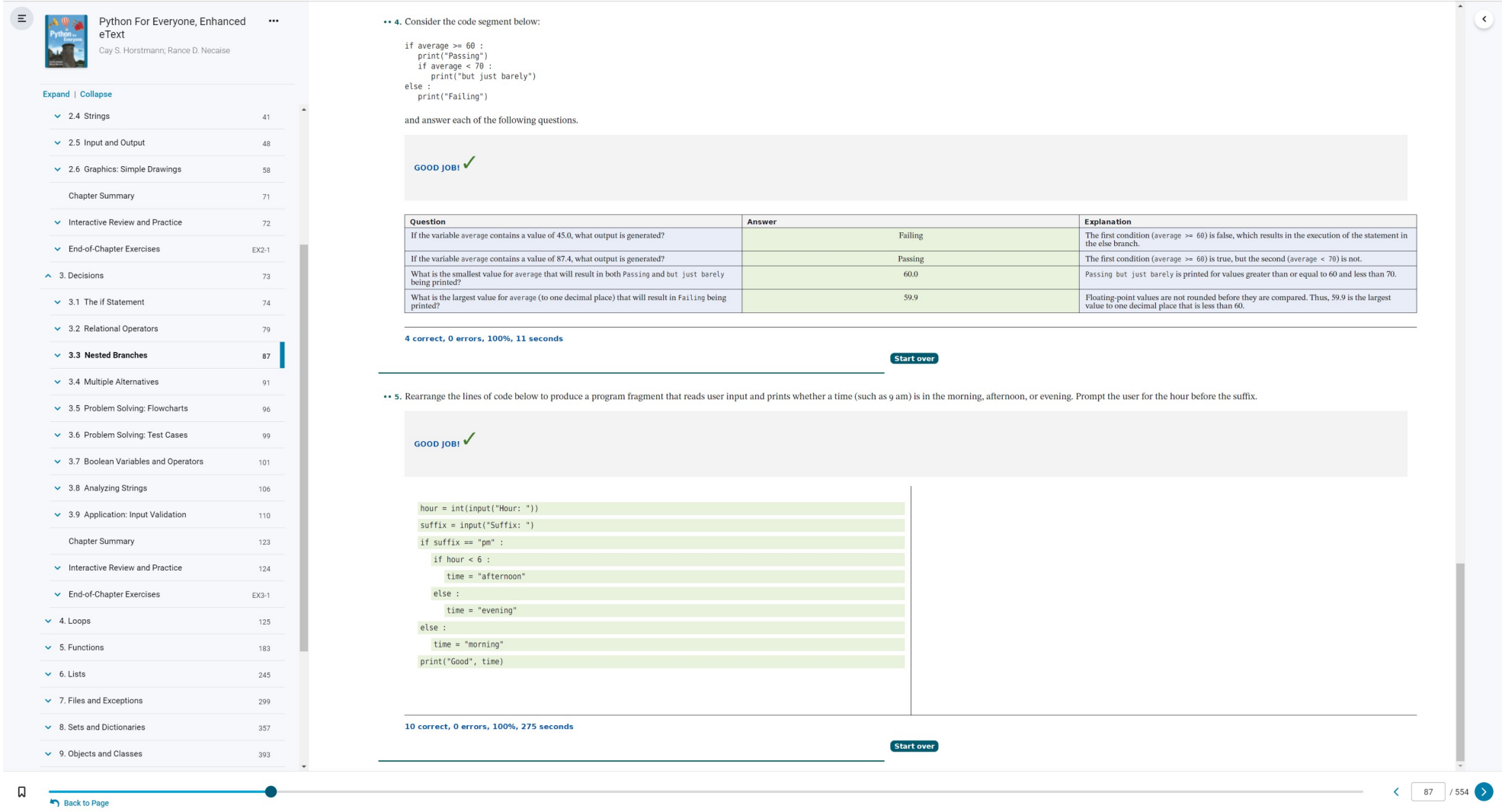
4 correct, 0 errors, 100%, 11 seconds

Start over

•• 5. Rearrange the lines of code below to produce a program fragment that reads user input and prints whether a time (such as 9 am) is in the morning, afternoon, or evening. Prompt the user for the hour before the suffix.

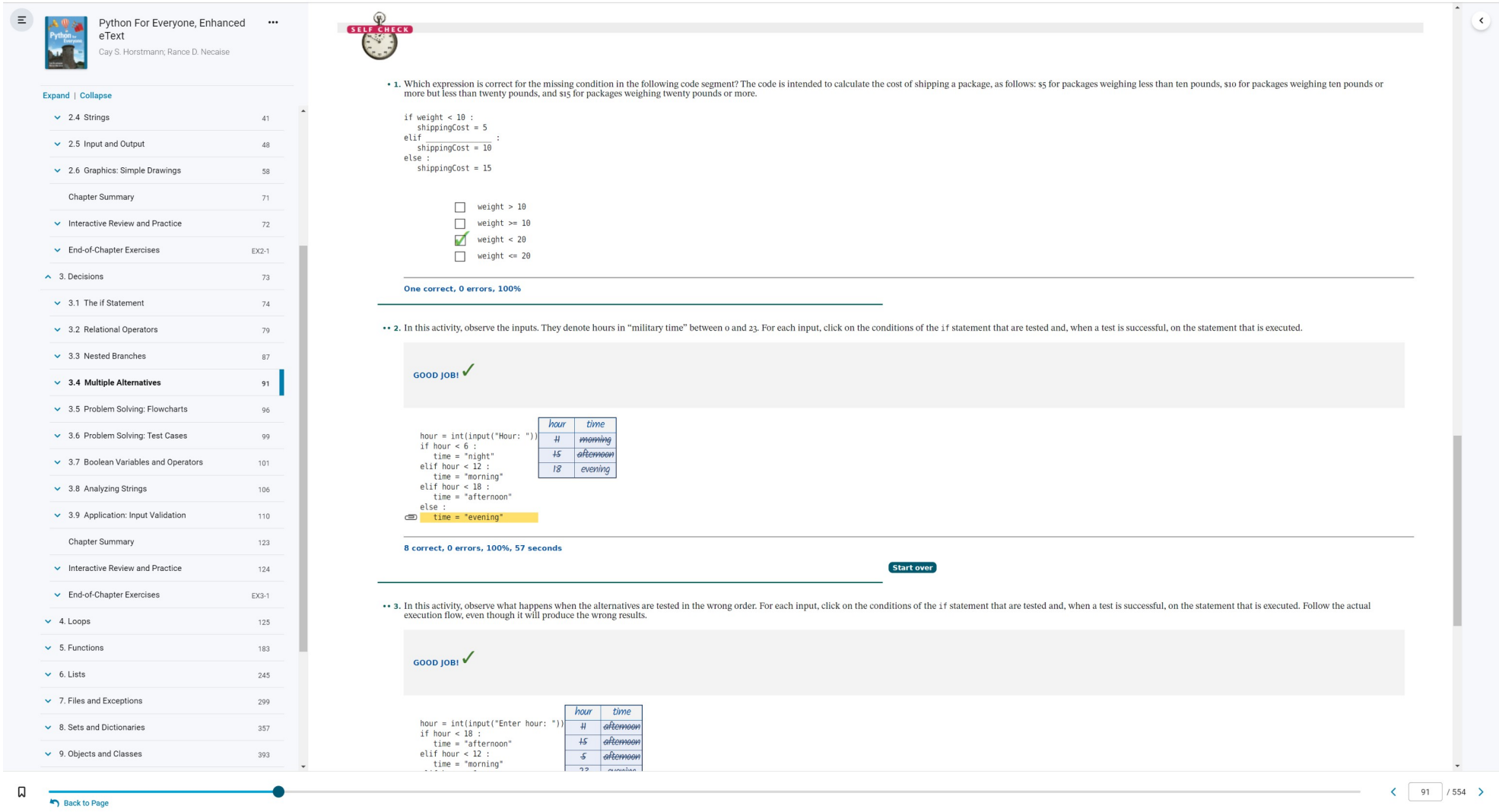
GOOD JOB! ✓

3.3 (p.2)

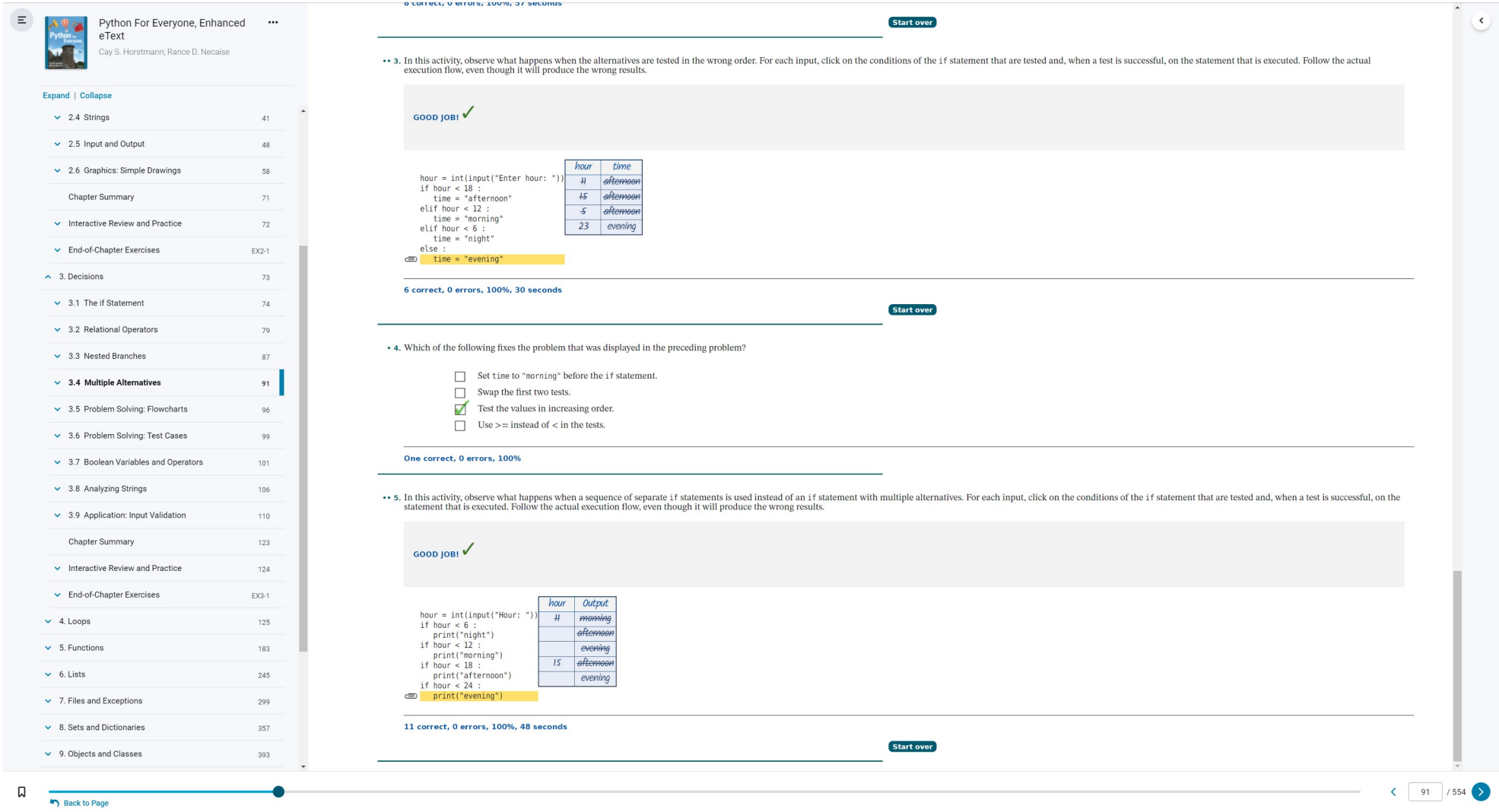

Python For Everyone, Enhanced eText
Cay S. Horstmann; Rance D. Necaise
Expand | Collapse
2.4 Strings 41
2.5 Input and Output 48
2.6 Graphics: Simple Drawings 58
Chapter Summary 71
Interactive Review and Practice 72
End-of-Chapter Exercises EX2-1
3. Decisions 73
3.1 The if Statement 74
3.2 Relational Operators 79
3.3 Nested Branches 87
3.4 Multiple Alternatives 91
3.5 Problem Solving: Flowcharts 96
3.6 Problem Solving: Test Cases 99
3.7 Boolean Variables and Operators 101
3.8 Analyzing Strings 106
3.9 Application: Input Validation 110
Chapter Summary 123
Interactive Review and Practice 124
End-of-Chapter Exercises EX3-1
4. Loops 125
5. Functions 183
6. Lists 245
7. Files and Exceptions 299
8. Sets and Dictionaries 357
9. Objects and Classes 393
•• 4. Consider the code segment below:
if average >= 60 :
print("Passing")
if average < 70 :
print("but just barely")
else :
print("Failing")
and answer each of the following questions.
GOOD JOB!
Question
Answer
Explanation
If the variable average contains a value of 45.0, what output is generated?
Failing
The first condition (average >= 60) is false, which results in the execution of the statement in the else branch.
If the variable average contains a value of 87.4, what output is generated?
Passing
The first condition (average >= 60) is true, but the second (average < 70) is not.
What is the smallest value for average that will result in both Passing and but just barely being printed?
60.0
Passing but just barely is printed for values greater than or equal to 60 and less than 70.
What is the largest value for average (to one decimal place) that will result in Failing being printed?
59.9
Floating-point values are not rounded before they are compared. Thus, 59.9 is the largest value to one decimal place that is less than 60.
4 correct, 0 errors, 100%, 11 seconds
Start over
•• 5. Rearrange the lines of code below to produce a program fragment that reads user input and prints whether a time (such as 9 am) is in the morning, afternoon, or evening. Prompt the user for the hour before the suffix.
GOOD JOB!
hour = int(input("Hour: "))
suffix = input("Suffix: ")
if suffix == "pm" :
if hour < 6 :
time = "afternoon"
else :
time = "evening"
else :
time = "morning"
print("Good", time)
10 correct, 0 errors, 100%, 275 seconds
Start over
Back to Page
87 / 554

3.3 (p.3)

Albert Nguyen
CS131-47853 MW 06:00 PM – 08:30 PM
Chapter 3 Homework – 3.1 – 3.9



3.4 (p.1)


Python For Everyone, Enhanced eText
Cay S. Horstmann; Rance D. Necaise
Expand | Collapse
2.4 Strings 41
2.5 Input and Output 48
2.6 Graphics: Simple Drawings 58
Chapter Summary 71
Interactive Review and Practice 72
End-of-Chapter Exercises EX2-1
3. Decisions 73
3.1 The if Statement 74
3.2 Relational Operators 79
3.3 Nested Branches 87
3.4 Multiple Alternatives 91
3.5 Problem Solving: Flowcharts 96
3.6 Problem Solving: Test Cases 99
3.7 Boolean Variables and Operators 101
3.8 Analyzing Strings 106
3.9 Application: Input Validation 110
Chapter Summary 123
Interactive Review and Practice 124
End-of-Chapter Exercises EX3-1
4. Loops 125
5. Functions 183
6. Lists 245
7. Files and Exceptions 299
8. Sets and Dictionaries 357
9. Objects and Classes 393
Start over
•• 3. In this activity, observe what happens when the alternatives are tested in the wrong order. For each input, click on the conditions of the if statement that are tested and, when a test is successful, on the statement that is executed. Follow the actual execution flow, even though it will produce the wrong results.
GOOD JOB!
hour = int(input("Enter hour: "))
if hour < 18 :
time = "afternoon"
elif hour < 12 :
time = "morning"
elif hour < 6 :
time = "night"
else :
time = "evening"
hour time
11 afternoon
15 afternoon
5 afternoon
23 evening
6 correct, 0 errors, 100%, 30 seconds
Start over
• 4. Which of the following fixes the problem that was displayed in the preceding problem?
Set time to "morning" before the if statement.
Swap the first two tests.
Test the values in increasing order.
Use >= instead of < in the tests.
One correct, 0 errors, 100%
•• 5. In this activity, observe what happens when a sequence of separate if statements is used instead of an if statement with multiple alternatives. For each input, click on the conditions of the if statement that are tested and, when a test is successful, on the statement that is executed. Follow the actual execution flow, even though it will produce the wrong results.
GOOD JOB!
hour = int(input("Hour: "))
if hour < 6 :
print("night")
if hour < 12 :
print("morning")
if hour < 18 :
print("afternoon")
if hour < 24 :
print("evening")
hour Output
11 morning
afternoon
evening
15 afternoon
evening
11 correct, 0 errors, 100%, 48 seconds
Start over
Back to Page
91 / 554

3.4 (p.2)

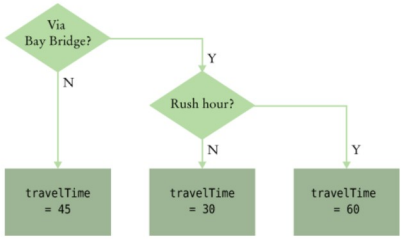
SELF CHECK

• **1.** In a flowchart, a decision is represented by

- [ ] a rectangle.
- [ ] a square.
- [x] a diamond with two outcomes.
- [ ] a diamond with three outcomes.

One correct, 0 errors, 100%

• **2.** Consider this flow chart for computing travel time between two towns.



Which code segment represents this flow chart?

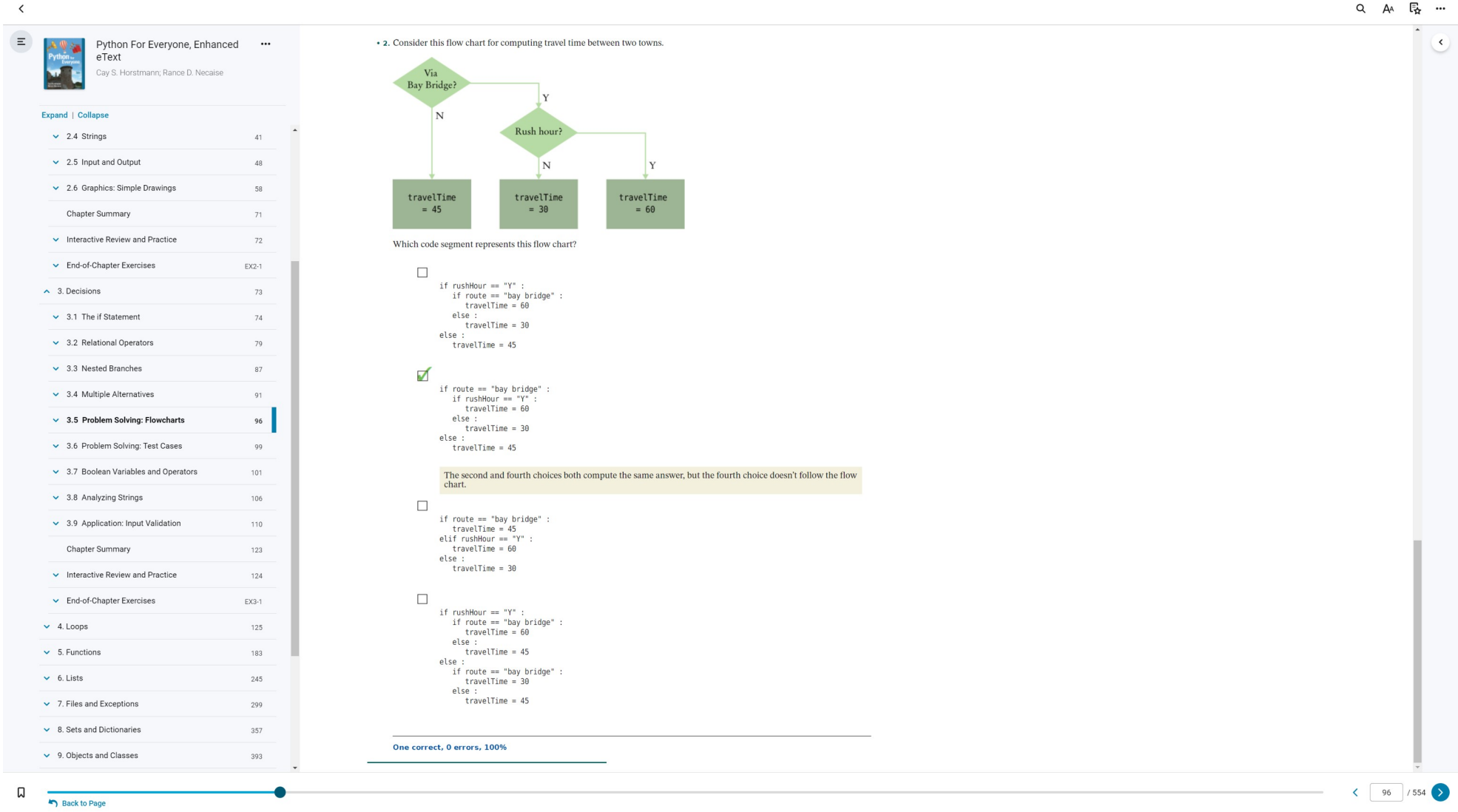
- [ ]
```
if rushHour == "Y" :
   if route == "bay bridge" :
      travelTime = 60
   else :
      travelTime = 30
else :
   travelTime = 45
```

- [x]
```
if route == "bay bridge" :
   if rushHour == "Y" :
      travelTime = 60
   else :
      travelTime = 30
else :
   travelTime = 45
```

The second and fourth choices both compute the same answer, but the fourth choice doesn't follow the flow chart.

- [ ]
```
if route == "bay bridge" :
   travelTime = 45
```


3.5 (p.1)



Python For Everyone, Enhanced eText
Cay S. Horstmann; Rance D. Necaise

Expand | Collapse

• 2. Consider this flow chart for computing travel time between two towns.

Which code segment represents this flow chart?

```
if rushHour == "Y" :
   if route == "bay bridge" :
      travelTime = 60
   else :
      travelTime = 30
else :
   travelTime = 45
```

```
if route == "bay bridge" :
   if rushHour == "Y" :
      travelTime = 60
   else :
      travelTime = 30
else :
   travelTime = 45
```

The second and fourth choices both compute the same answer, but the fourth choice doesn't follow the flow chart.

```
if route == "bay bridge" :
   travelTime = 45
elif rushHour == "Y" :
   travelTime = 60
else :
   travelTime = 30
```

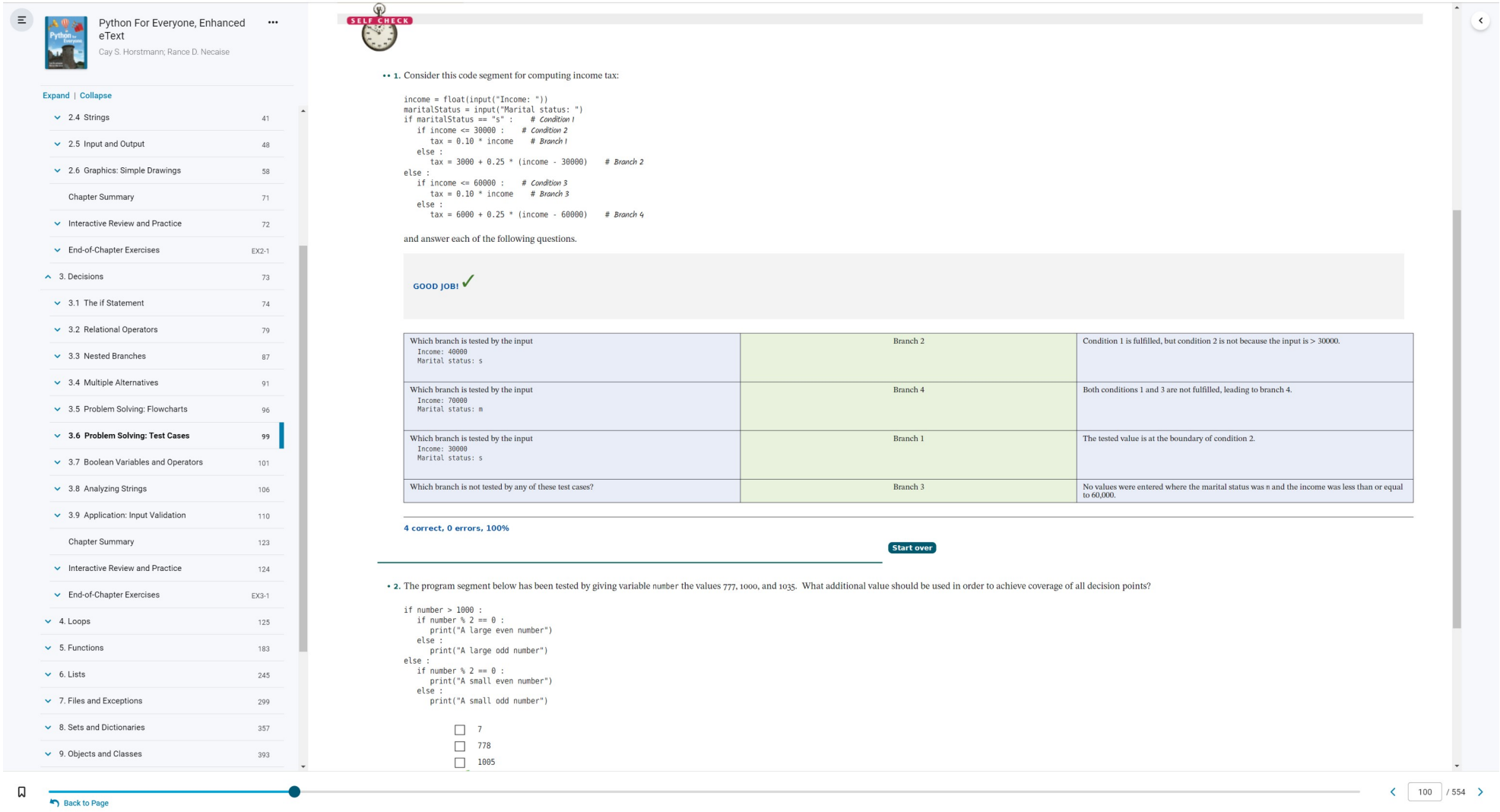
```
if rushHour == "Y" :
   if route == "bay bridge" :
      travelTime = 60
   else :
      travelTime = 45
else :
   if route == "bay bridge" :
      travelTime = 30
   else :
      travelTime = 45
```

One correct, 0 errors, 100%

Back to Page

3.5 (p.2)

Albert Nguyen
CS131-47853 MW 06:00 PM – 08:30 PM
Chapter 3 Homework – 3.1 – 3.9



3.6 (p.1)

GOOD JOB! ✓

Which branch is tested by the input
`Income: 40000`
`Marital status: s`
	Branch 2	Condition 1 is fulfilled, but condition 2 is not because the input is > 30000.

Which branch is tested by the input
`Income: 70000`
`Marital status: m`
	Branch 4	Both conditions 1 and 3 are not fulfilled, leading to branch 4.

Which branch is tested by the input
`Income: 30000`
`Marital status: s`
	Branch 1	The tested value is at the boundary of condition 2.

Which branch is not tested by any of these test cases?	Branch 3	No values were entered where the marital status was `m` and the income was less than or equal to 60,000.

4 correct, 0 errors, 100%

Start over

• 2. The program segment below has been tested by giving variable `number` the values 777, 1000, and 1035. What additional value should be used in order to achieve coverage of all decision points?

```
if number > 1000 :
   if number % 2 == 0 :
      print("A large even number")
   else :
      print("A large odd number")
else :
   if number % 2 == 0 :
      print("A small even number")
   else :
      print("A small odd number")
```

- ☐ 7
- ☐ 778
- ☐ 1005
- ☑ 1006

One correct, 0 errors, 100%

•• 3. Consider the elevator simulation program from Section 3.1. Which of the following inputs would be an appropriate set of test cases?

- ☐ 11 and 12
- ☐ 14 and 15
- ☑ 12 and 14

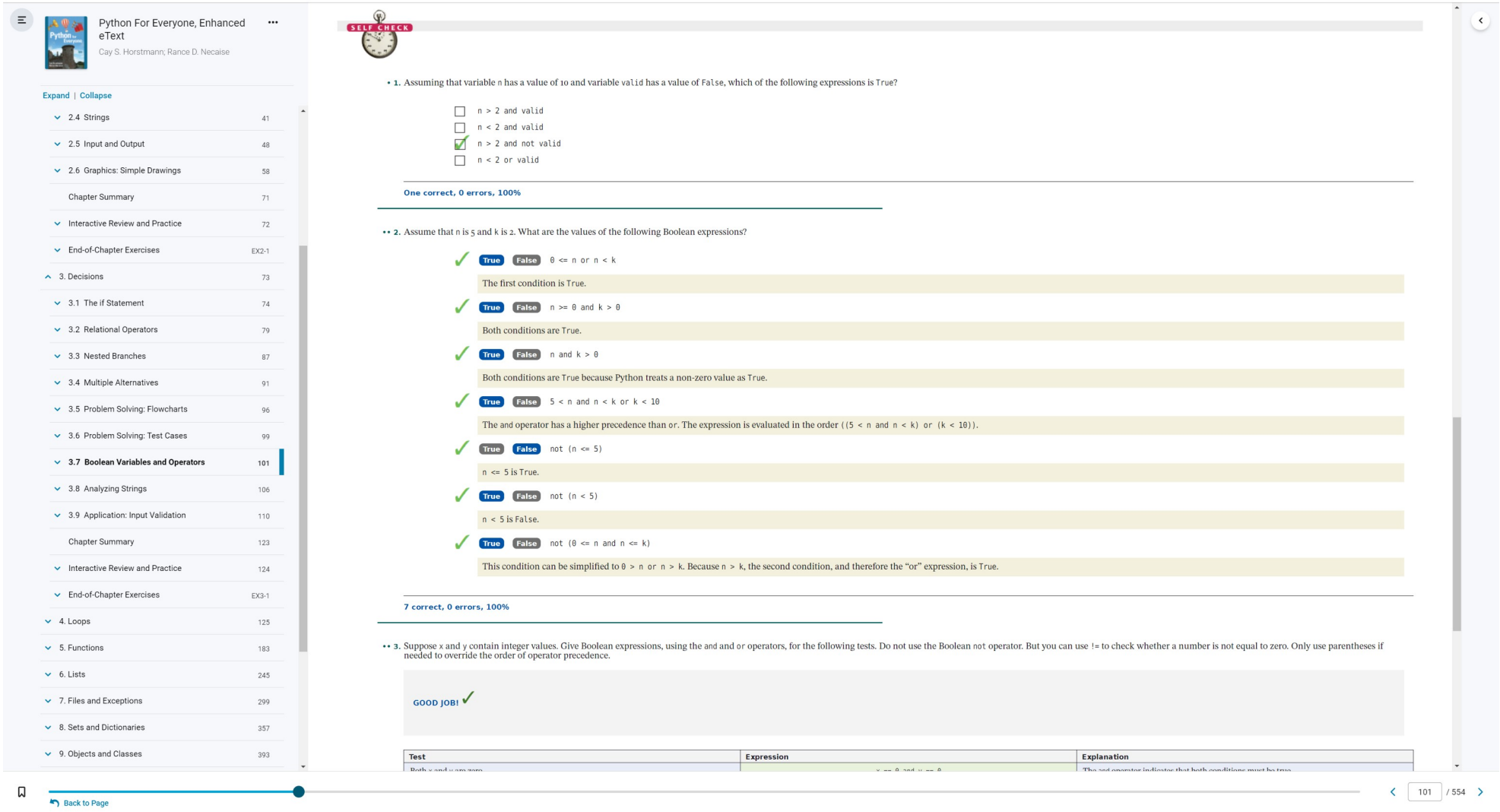
  You want to test floors above and below floor 13. The specification isn't clear what should happen with an input of 13. In Section 3.9, you will see a version of the program that reports an error for that input.

- ☐ 12, 13, 14

One correct, 0 errors, 100%

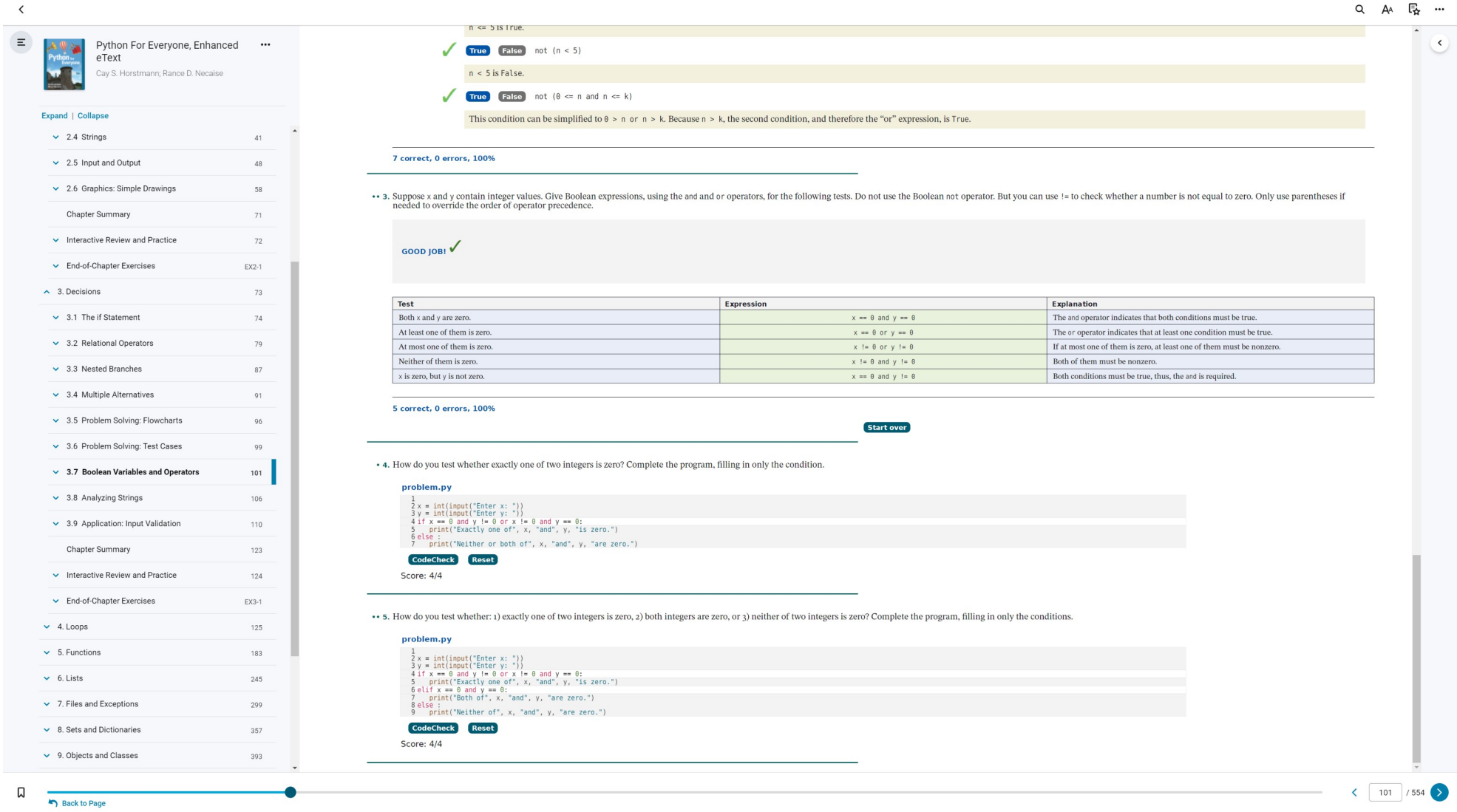
3.6 (p.2)

Albert Nguyen
CS131-47853 MW 06:00 PM – 08:30 PM
Chapter 3 Homework – 3.1 – 3.9



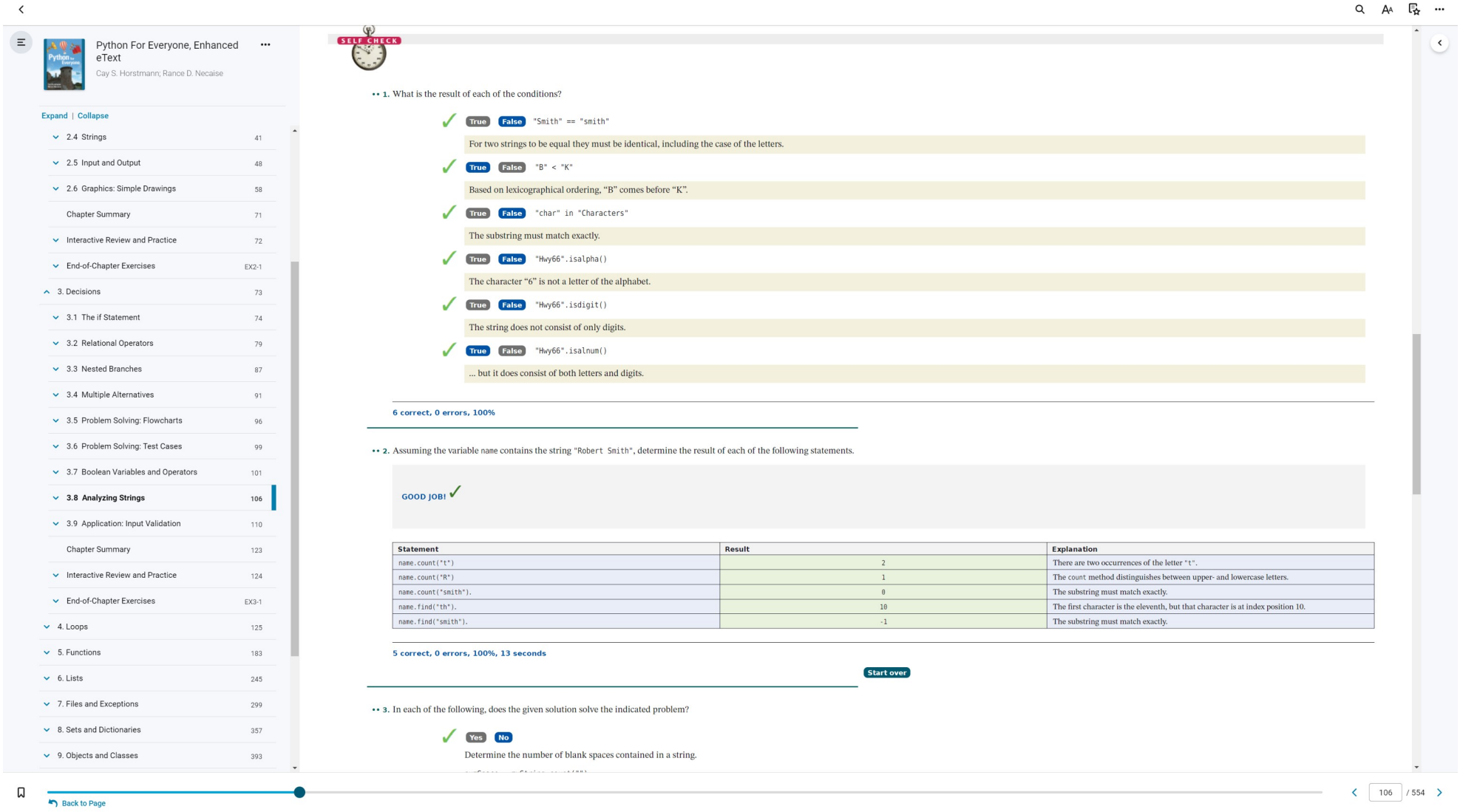
3.7 (p.1)

Albert Nguyen
CS131-47853 MW 06:00 PM – 08:30 PM
Chapter 3 Homework – 3.1 – 3.9

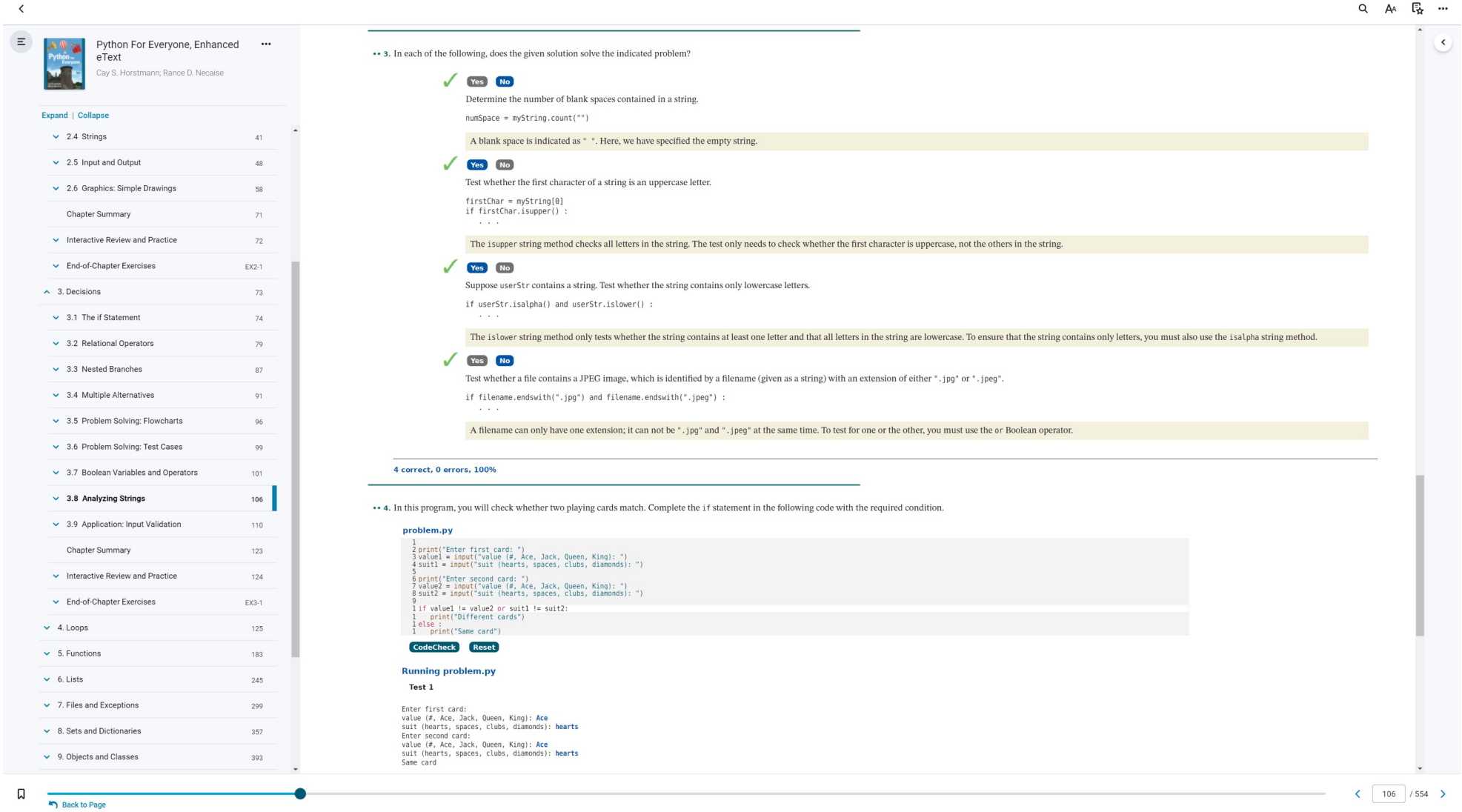


3.7 (p.2)

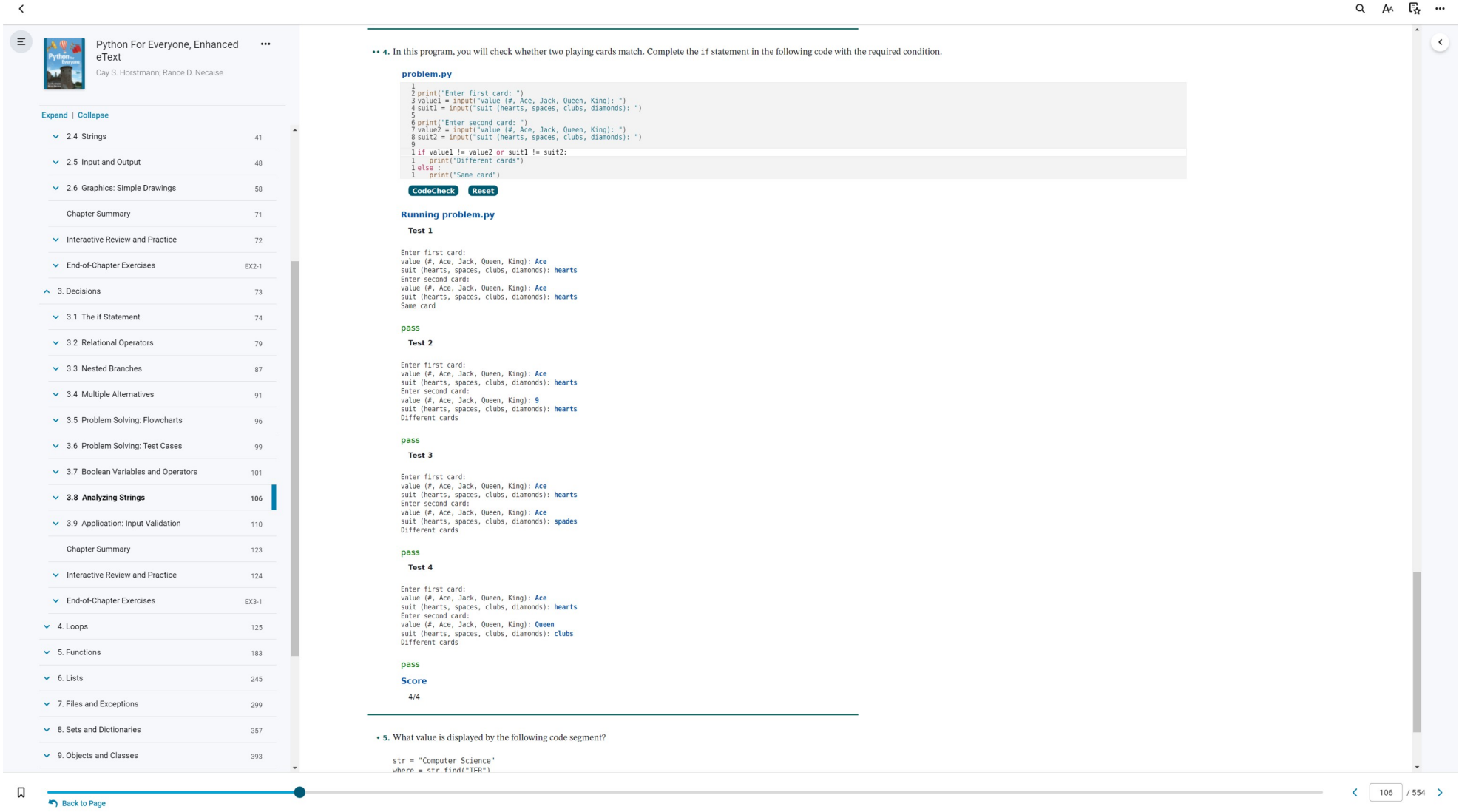
Albert Nguyen
CS131-47853 MW 06:00 PM – 08:30 PM
Chapter 3 Homework – 3.1 – 3.9



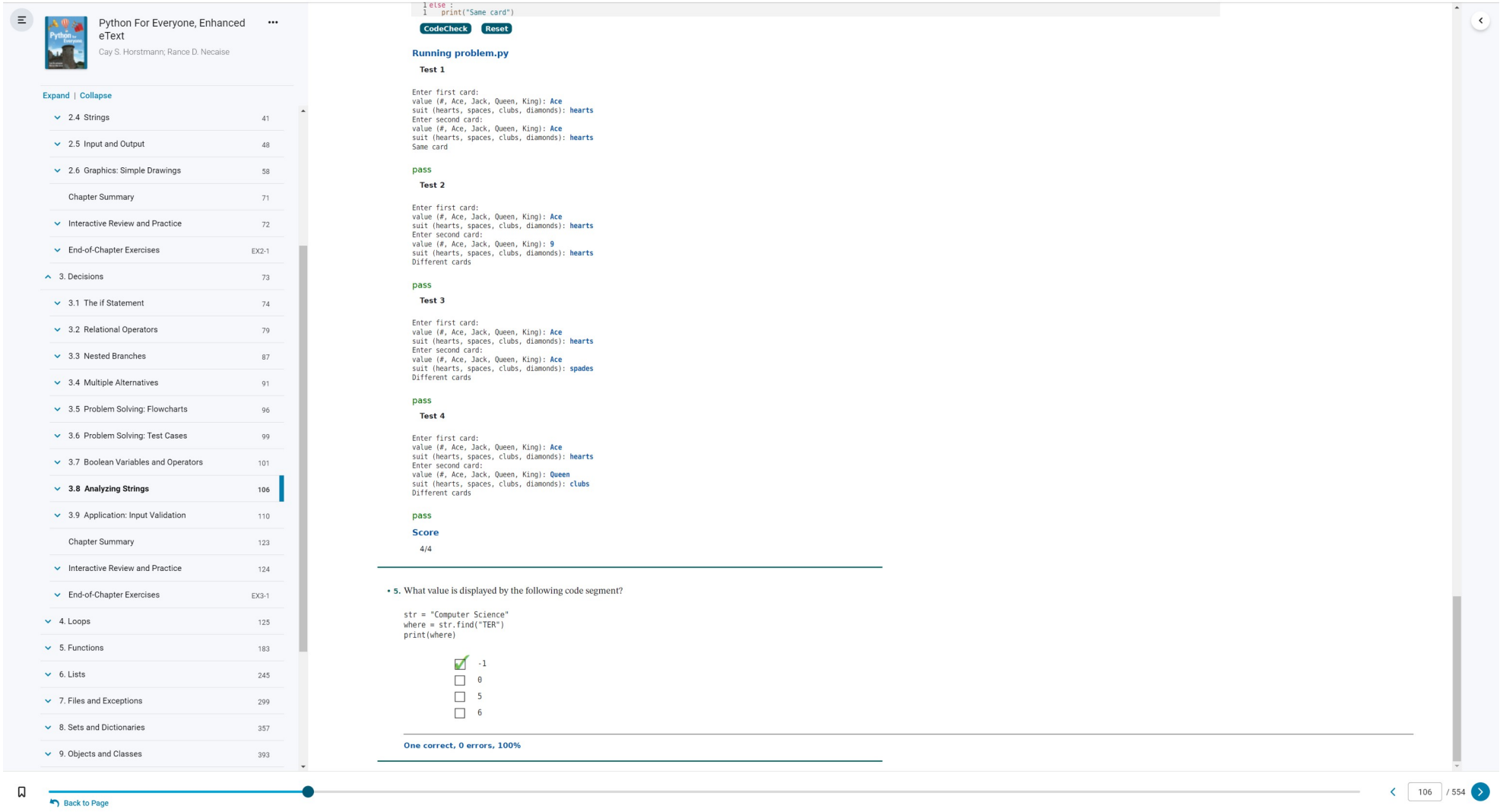
3.8 (p.1)


Python For Everyone, Enhanced eText
Cay S. Horstmann; Rance D. Necaise
Expand | Collapse
2.4 Strings 41
2.5 Input and Output 48
2.6 Graphics: Simple Drawings 58
Chapter Summary 71
Interactive Review and Practice 72
End-of-Chapter Exercises EX2-1
3. Decisions 73
3.1 The if Statement 74
3.2 Relational Operators 79
3.3 Nested Branches 87
3.4 Multiple Alternatives 91
3.5 Problem Solving: Flowcharts 96
3.6 Problem Solving: Test Cases 99
3.7 Boolean Variables and Operators 101
3.8 Analyzing Strings 106
3.9 Application: Input Validation 110
Chapter Summary 123
Interactive Review and Practice 124
End-of-Chapter Exercises EX3-1
4. Loops 125
5. Functions 183
6. Lists 245
7. Files and Exceptions 299
8. Sets and Dictionaries 357
9. Objects and Classes 393
•• 3. In each of the following, does the given solution solve the indicated problem?
Yes No
Determine the number of blank spaces contained in a string.
numSpace = myString.count("")
A blank space is indicated as " ". Here, we have specified the empty string.
Yes No
Test whether the first character of a string is an uppercase letter.
firstChar = myString[0]
if firstChar.isupper() :
. . .
The isupper string method checks all letters in the string. The test only needs to check whether the first character is uppercase, not the others in the string.
Yes No
Suppose userStr contains a string. Test whether the string contains only lowercase letters.
if userStr.isalpha() and userStr.islower() :
. . .
The islower string method only tests whether the string contains at least one letter and that all letters in the string are lowercase. To ensure that the string contains only letters, you must also use the isalpha string method.
Yes No
Test whether a file contains a JPEG image, which is identified by a filename (given as a string) with an extension of either ".jpg" or ".jpeg".
if filename.endswith(".jpg") and filename.endswith(".jpeg") :
. . .
A filename can only have one extension; it can not be ".jpg" and ".jpeg" at the same time. To test for one or the other, you must use the or Boolean operator.
4 correct, 0 errors, 100%
•• 4. In this program, you will check whether two playing cards match. Complete the if statement in the following code with the required condition.
problem.py
print("Enter first card: ")
value1 = input("value (#, Ace, Jack, Queen, King): ")
suit1 = input("suit (hearts, spaces, clubs, diamonds): ")
print("Enter second card: ")
value2 = input("value (#, Ace, Jack, Queen, King): ")
suit2 = input("suit (hearts, spaces, clubs, diamonds): ")
if value1 != value2 or suit1 != suit2:
print("Different cards")
else :
print("Same card")
CodeCheck Reset
Running problem.py
Test 1
Enter first card:
value (#, Ace, Jack, Queen, King): Ace
suit (hearts, spaces, clubs, diamonds): hearts
Enter second card:
value (#, Ace, Jack, Queen, King): Ace
suit (hearts, spaces, clubs, diamonds): hearts
Same card
Back to Page
106 / 554

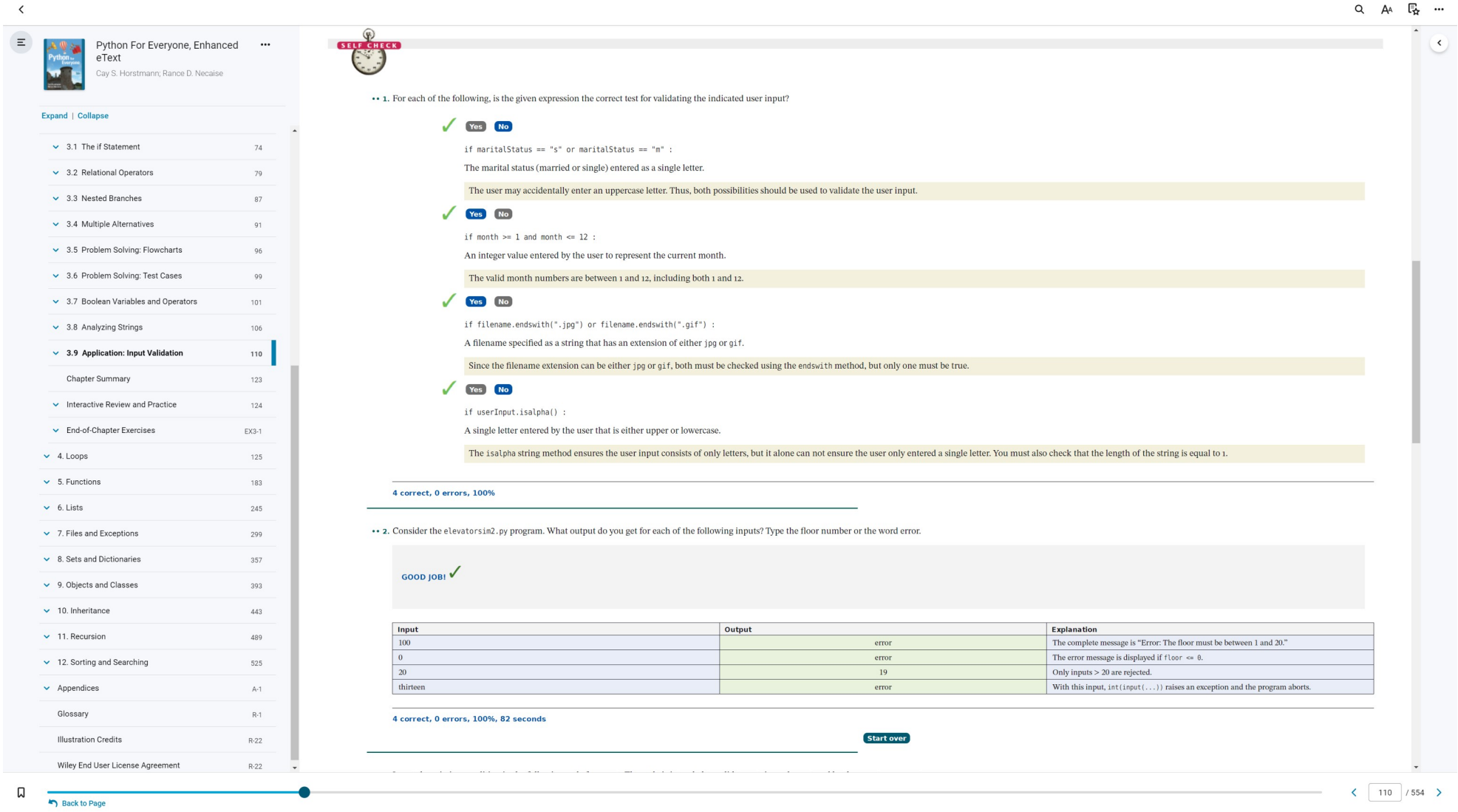
3.8 (p.2)


Python For Everyone, Enhanced eText
Cay S. Horstmann; Rance D. Necaise
Expand | Collapse
2.4 Strings 41
2.5 Input and Output 48
2.6 Graphics: Simple Drawings 58
Chapter Summary 71
Interactive Review and Practice 72
End-of-Chapter Exercises EX2-1
3. Decisions 73
3.1 The if Statement 74
3.2 Relational Operators 79
3.3 Nested Branches 87
3.4 Multiple Alternatives 91
3.5 Problem Solving: Flowcharts 96
3.6 Problem Solving: Test Cases 99
3.7 Boolean Variables and Operators 101
3.8 Analyzing Strings 106
3.9 Application: Input Validation 110
Chapter Summary 123
Interactive Review and Practice 124
End-of-Chapter Exercises EX3-1
4. Loops 125
5. Functions 183
6. Lists 245
7. Files and Exceptions 299
8. Sets and Dictionaries 357
9. Objects and Classes 393
•• 4. In this program, you will check whether two playing cards match. Complete the if statement in the following code with the required condition.
problem.py
1
2 print("Enter first card: ")
3 value1 = input("value (#, Ace, Jack, Queen, King): ")
4 suit1 = input("suit (hearts, spaces, clubs, diamonds): ")
5
6 print("Enter second card: ")
7 value2 = input("value (#, Ace, Jack, Queen, King): ")
8 suit2 = input("suit (hearts, spaces, clubs, diamonds): ")
9
10 if value1 != value2 or suit1 != suit2:
11 print("Different cards")
12 else :
13 print("Same card")
CodeCheck
Reset
Running problem.py
Test 1
Enter first card:
value (#, Ace, Jack, Queen, King): Ace
suit (hearts, spaces, clubs, diamonds): hearts
Enter second card:
value (#, Ace, Jack, Queen, King): Ace
suit (hearts, spaces, clubs, diamonds): hearts
Same card
pass
Test 2
Enter first card:
value (#, Ace, Jack, Queen, King): Ace
suit (hearts, spaces, clubs, diamonds): hearts
Enter second card:
value (#, Ace, Jack, Queen, King): 9
suit (hearts, spaces, clubs, diamonds): hearts
Different cards
pass
Test 3
Enter first card:
value (#, Ace, Jack, Queen, King): Ace
suit (hearts, spaces, clubs, diamonds): hearts
Enter second card:
value (#, Ace, Jack, Queen, King): Ace
suit (hearts, spaces, clubs, diamonds): spades
Different cards
pass
Test 4
Enter first card:
value (#, Ace, Jack, Queen, King): Ace
suit (hearts, spaces, clubs, diamonds): hearts
Enter second card:
value (#, Ace, Jack, Queen, King): Queen
suit (hearts, spaces, clubs, diamonds): clubs
Different cards
pass
Score
4/4
• 5. What value is displayed by the following code segment?
str = "Computer Science"
where = str.find("TER")
Back to Page
106 / 554

3.8 (p.3)


Python For Everyone, Enhanced eText
Cay S. Horstmann; Rance D. Necaise
Expand | Collapse
2.4 Strings 41
2.5 Input and Output 48
2.6 Graphics: Simple Drawings 58
Chapter Summary 71
Interactive Review and Practice 72
End-of-Chapter Exercises EX2-1
3. Decisions 73
3.1 The if Statement 74
3.2 Relational Operators 79
3.3 Nested Branches 87
3.4 Multiple Alternatives 91
3.5 Problem Solving: Flowcharts 96
3.6 Problem Solving: Test Cases 99
3.7 Boolean Variables and Operators 101
3.8 Analyzing Strings 106
3.9 Application: Input Validation 110
Chapter Summary 123
Interactive Review and Practice 124
End-of-Chapter Exercises EX3-1
4. Loops 125
5. Functions 183
6. Lists 245
7. Files and Exceptions 299
8. Sets and Dictionaries 357
9. Objects and Classes 393
else :
print("Same card")
CodeCheck
Reset
Running problem.py
Test 1
Enter first card:
value (#, Ace, Jack, Queen, King): Ace
suit (hearts, spaces, clubs, diamonds): hearts
Enter second card:
value (#, Ace, Jack, Queen, King): Ace
suit (hearts, spaces, clubs, diamonds): hearts
Same card
pass
Test 2
Enter first card:
value (#, Ace, Jack, Queen, King): Ace
suit (hearts, spaces, clubs, diamonds): hearts
Enter second card:
value (#, Ace, Jack, Queen, King): 9
suit (hearts, spaces, clubs, diamonds): hearts
Different cards
pass
Test 3
Enter first card:
value (#, Ace, Jack, Queen, King): Ace
suit (hearts, spaces, clubs, diamonds): hearts
Enter second card:
value (#, Ace, Jack, Queen, King): Ace
suit (hearts, spaces, clubs, diamonds): spades
Different cards
pass
Test 4
Enter first card:
value (#, Ace, Jack, Queen, King): Ace
suit (hearts, spaces, clubs, diamonds): hearts
Enter second card:
value (#, Ace, Jack, Queen, King): Queen
suit (hearts, spaces, clubs, diamonds): clubs
Different cards
pass
Score
4/4
• 5. What value is displayed by the following code segment?
str = "Computer Science"
where = str.find("TER")
print(where)
-1
0
5
6
One correct, 0 errors, 100%
Back to Page
106 / 554

3.8 (p.4)



Python For Everyone, Enhanced eText
Cay S. Horstmann; Rance D. Necaise

Expand | Collapse

SELF CHECK

•• 1. For each of the following, is the given expression the correct test for validating the indicated user input?

✓ Yes No

```
if maritalStatus == "s" or maritalStatus == "m" :
```

The marital status (married or single) entered as a single letter.

The user may accidentally enter an uppercase letter. Thus, both possibilities should be used to validate the user input.

✓ Yes No

```
if month >= 1 and month <= 12 :
```

An integer value entered by the user to represent the current month.

The valid month numbers are between 1 and 12, including both 1 and 12.

✓ Yes No

```
if filename.endswith(".jpg") or filename.endswith(".gif") :
```

A filename specified as a string that has an extension of either `jpg` or `gif`.

Since the filename extension can be either `jpg` or `gif`, both must be checked using the `endswith` method, but only one must be true.

✓ Yes No

```
if userInput.isalpha() :
```

A single letter entered by the user that is either upper or lowercase.

The `isalpha` string method ensures the user input consists of only letters, but it alone can not ensure the user only entered a single letter. You must also check that the length of the string is equal to 1.

4 correct, 0 errors, 100%

•• 2. Consider the `elevatorsim2.py` program. What output do you get for each of the following inputs? Type the floor number or the word error.

GOOD JOB! ✓

Input	Output	Explanation
100	error	The complete message is "Error: The floor must be between 1 and 20."
0	error	The error message is displayed if `floor <= 0`.
20	19	Only inputs > 20 are rejected.
thirteen	error	With this input, `int(input(...))` raises an exception and the program aborts.

4 correct, 0 errors, 100%, 82 seconds

Start over

Back to Page

110 / 554

3.9 (p.1)

Albert Nguyen
CS131-47853 MW 06:00 PM – 08:30 PM
Chapter 3 Homework – 3.1 – 3.9

•• 3. Insert the missing condition in the following code fragment. The code is intended to validate a string value entered by the user to represent a yes or no response.

```
if _____ :
   valid = True
else :
   valid = False
```

- [ ] `response == "y" or response == "n"`
- [ ] `response == "y" and response == "n"`
- [x] `response == "y" or response == "Y" or response =="n" or response == "N"`

  If the user is not instructed to specifically enter a lowercase or uppercase letter for the response, so you must check for both cases.
- [ ] `response == "y" or response == "Y" and response == "n" or response == "N"`

One correct, 0 errors, 100%

•• 4. In the Sherlock Holmes story "The Adventure of the Sussex Vampire", the inimitable detective uttered these words: "Matilda Briggs was not the name of a young woman, Watson, ... It was a ship which is associated with the giant rat of Sumatra, a story for which the world is not yet prepared." Over a hundred years later, researchers found giant rats in Western New Guinea, another part of Indonesia.

© jeanma85/iStockphoto.

Suppose you are charged with writing a program that processes rat weights. We don't know for sure how giant a rat could be, but the New Guinea rats weighed no more than 2 kg. A regular house rat (rattus rattus) weighs up to 0.2 kg, so we'll say that any weight > 10 kg was surely an input error, perhaps confusing grams and kilograms. Of course, inputs should be positive numbers. Rearrange the following lines of code to supply error handling. Not all lines are useful.

GOOD JOB! ✓

```
weight = float(input("Enter weight in kg: "))
if weight <= 0 :
   print("Error: Weight must be positive.")
elif weight > 10 :
   print("Error: Weight > 10 kg.")
else :
   ## Process weight . . .
```

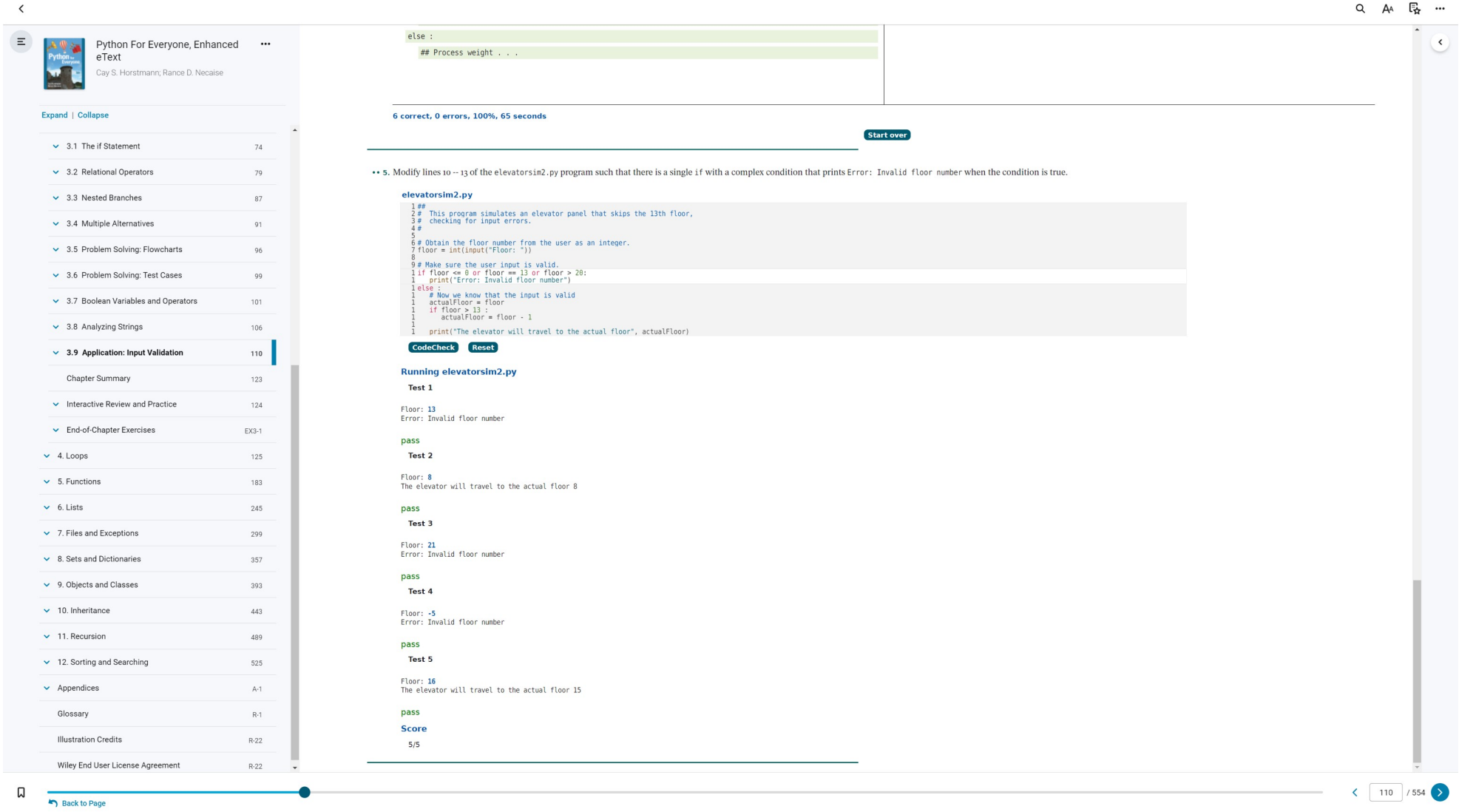
```
if weight >= 0 :
elif weight < 10 :
```

6 correct, 0 errors, 100%, 65 seconds

Start over

•• 5. Modify lines 10 -- 13 of the `elevatorsim2.py` program such that there is a single `if` with a complex condition that prints `Error: Invalid floor number` when the condition is true.

3.9 (p.2)


Python For Everyone, Enhanced eText
Cay S. Horstmann; Rance D. Necaise
Expand | Collapse
3.1 The if Statement 74
3.2 Relational Operators 79
3.3 Nested Branches 87
3.4 Multiple Alternatives 91
3.5 Problem Solving: Flowcharts 96
3.6 Problem Solving: Test Cases 99
3.7 Boolean Variables and Operators 101
3.8 Analyzing Strings 106
3.9 Application: Input Validation 110
Chapter Summary 123
Interactive Review and Practice 124
End-of-Chapter Exercises EX3-1
4. Loops 125
5. Functions 183
6. Lists 245
7. Files and Exceptions 299
8. Sets and Dictionaries 357
9. Objects and Classes 393
10. Inheritance 443
11. Recursion 489
12. Sorting and Searching 525
Appendices A-1
Glossary R-1
Illustration Credits R-22
Wiley End User License Agreement R-22
else :
Process weight . . .
6 correct, 0 errors, 100%, 65 seconds
Start over
•• 5. Modify lines 10 -- 13 of the elevatorsim2.py program such that there is a single if with a complex condition that prints Error: Invalid floor number when the condition is true.
elevatorsim2.py
This program simulates an elevator panel that skips the 13th floor, # checking for input errors. # # Obtain the floor number from the user as an integer. floor = int(input("Floor: ")) # Make sure the user input is valid. if floor <= 0 or floor == 13 or floor > 20: print("Error: Invalid floor number") else : # Now we know that the input is valid actualFloor = floor if floor > 13 : actualFloor = floor - 1 print("The elevator will travel to the actual floor", actualFloor)
CodeCheck Reset
Running elevatorsim2.py
Test 1
Floor: 13
Error: Invalid floor number
pass
Test 2
Floor: 8
The elevator will travel to the actual floor 8
pass
Test 3
Floor: 21
Error: Invalid floor number
pass
Test 4
Floor: -5
Error: Invalid floor number
pass
Test 5
Floor: 16
The elevator will travel to the actual floor 15
pass
Score
5/5
Back to Page
110 / 554

3.9 (p.3)