Conditional Statements
8. Boolean Expressions for Conditions
9. Quiz: Boolean Expressions for Conditions
10. Solution: Boolean Expressions for Conditions
11. For Loops
12. Practice: For Loops
13. Solution: For Loops Practice
Downloadable resources

Conditional Statements
8. Boolean Expressions for Conditions
9. Quiz: Boolean Expressions for Conditions
10. Solution: Boolean Expressions for Conditions
11. For Loops

Solution: Boolean Expressions for Conditions

Quiz Solution: Evaluate composed boolean expressions

```
altitude < 1000 and speed > 100
```

`altitude < 1000` is `False`, so we don't even need to check the second condition - the whole expression is `False`.

```
(propulsion == "Jet" or propulsion == "Turboprop") and speed < 300 and altitude > 20000
```

`propulsion == "Jet"` is `False`, and `propulsion == "Turboprop"` is `False`, so the whole expression inside the parentheses is `False`. It is combined with the other expressions with `and`, so we don't even need to check these - the whole expression must be `False` because the first part is `False`.

```
not (speed > 400 and propulsion == "Propeller")
```

To work this one out, we need to look at the inside of the parentheses first, then apply `not` to that. `speed > 400` is False, and because we are using `and` this makes the whole of the expression inside the parentheses `False`. Applying `not` reverses this, so this expression is `True`.

```
(altitude > 500 and speed > 100) or not propulsion == "Propeller"
```

Let's start by looking inside the parentheses. `altitude > 500` is `True`, and `speed` is greater than `100`, so the expression inside the parenthesis is `True`. Whatever the value of the other expression, because they are connected by `or`, the whole expression will evaluate to `True`.

Quiz Solution: Using Truth Values of Objects

```
points = 174

points = 174  # use this input when submitting your answer

## set prize to default value of None
prize = None

## use the value of points to assign prize to the correct prize name
if points <= 50:
    prize = "wooden rabbit"
elif 151 <= points <= 180:
    prize = "wafer-thin mint"
elif points >= 181:
    prize = "penguin"

## use the truth value of prize to assign result to the correct message
if prize:
    result = "Congratulations! You won a {}!".format(prize)
else:
    result = "Oh dear, no prize this time."

print(result)
```

Output

```
Congratulations! You won a wafer-thin mint!
```

We first set `prize` to `None` and then update it only if falls into a bracket that results in winning a prize. This is accomplished in the first `if` statement. We then use the truth value of `prize` to assign `result` to a message based on whether a prize was won.

Remember when `prize = "penguin"`, or any other non-empty string, then the `if prize` condition is True!

Previous

Next

Give Page Feedback