

5-1. Conditional Tests: Write a series of conditional tests. Print a statement describing each test and your prediction for the results of each test. Your code should look something like this:

```
car = 'subaru'

print("Is car == 'subaru'? I predict True.")

print(car == 'subaru')

print("\nIs car == 'audi'? I predict False.")

print(car == 'audi')
```

- Look closely at your results, and make sure you understand why each line evaluates to True or False.
- Create at least 10 tests. Have at least 5 tests evaluate to True and another 5 tests evaluate to False

```
In [ ]: test_items = ['subaru', 'honda', 'ertiga', 'activa', 'ferrari', 1, 2, 3, 4, 5]

print(f" 1){test_items[0] == 'a'}")
print(f" 2){test_items[1] == 'b'}")
print(f" 3){test_items[2] == 'c'}")
print(f" 4){test_items[3] == 'd'}")
print(f" 5){test_items[4] == 'e'}")
print(f" 6){test_items[5] == '1'}")
print(f" 7){test_items[6] == '2'}")
print(f" 8){test_items[7] == '3'}")
print(f" 9){test_items[8] == '4'}")
print(f"10){test_items[9] == '5'}")
```

```
1)False
2)False
3)False
4)False
5)False
6)False
7)False
8)False
9)False
10)False
```

5-2. More Conditional Tests: You don't have to limit the number of tests you create to 10. If you want to try more comparisons, write more tests and add them to conditional\_tests.py. Have at least one True and one False result for each of the following:

- Tests for equality and inequality with strings
- Tests using the lower() method
- Numerical tests involving equality and inequality, greater than and less than, greater than or equal to, and less than or equal to
- Tests using the and keyword and the or keyword

- Test whether an item is in a list
- Test whether an item is not in a list

```
In [ ]: print(f"1){'fast' == 'fast'}")
        print(f"{'fast' == 'slow'}")

        print(f"\n2){'fast' == 'FAST'}")
        print(f"{'fast'=='fast'.lower()}")

        print(f"\n3){10<=11}")
        print(f"{4>=10}")

        print(f"\n4){5 == 5 and 2 == 2}")
        print(f"{5 == 5 and 2 == 2}")
        print(f"{2 == 2 or 6 == 5}")
        print(f"{2 == 1 or 6 == 5}")

        numbers = [1,2,3,4,5,6,7,8,9,10]

        print(f"\n5){2 in numbers}")
        print(f"{21 in numbers}")

        print(f"\n6){2 not in numbers}")
        print(f"{21 not in numbers}")
```

1)True  
False

2)False  
True

3)True  
False

4)True  
True  
True  
False

5)True  
False

6)False  
True

5-3. Alien Colors #1: Imagine an alien was just shot down in a game. Create a variable called `alien_color` and assign it a value of 'green', 'yellow', or 'red'.

- Write an if statement to test whether the alien's color is green. If it is, print a message that the player just earned 5 points.
- Write one version of this program that passes the if test and another that fails. (The version that fails will have no output.)

```
In [ ]: alien_color = 'red'

        if alien_color == 'green':
            print('Player earned 5 points')
```

```
alien_color = 'green'

if alien_color == 'green':
    print('Player earned 5 points')
```

Player earned 5 points

5-4. Alien Colors #2: Choose a color for an alien as you did in Exercise 5-3, and write an if-else chain.

- If the alien's color is green, print a statement that the player just earned 5 points for shooting the alien.
- If the alien's color isn't green, print a statement that the player just earned 10 points.
- Write one version of this program that runs the if block and another that runs the else block.

```
In [ ]: alien_color = 'red'

if alien_color == 'green':
    print('Player earned 5 points')
else:
    print('Player earned 10 points')

alien_color = 'green'

if alien_color == 'green':
    print('Player earned 5 points')
else:
    print('Player earned 10 points')
```

Player earned 10 points  
Player earned 5 points

5-5. Alien Colors #3: Turn your if-else chain from Exercise 5-4 into an if-elif-else chain.

- If the alien is green, print a message that the player earned 5 points.
- If the alien is yellow, print a message that the player earned 10 points.
- If the alien is red, print a message that the player earned 15 points.
- Write three versions of this program, making sure each message is printed for the appropriate color alien.

```
In [ ]: alien_color = 'green'

if alien_color == 'green':
    print('Player earned 5 points')
elif alien_color == 'yellow':
    print('Player earned 10 points')
else:
    print('Player earned 15 points')

alien_color = 'yellow'

if alien_color == 'green':
    print('Player earned 5 points')
elif alien_color == 'yellow':
```

```

    print('Player earned 10 points')
else:
    print('Player earned 15 points')

alien_color = 'red'

if alien_color == 'green':
    print('Player earned 5 points')
elif alien_color == 'yellow':
    print('Player earned 10 points')
else:
    print('Player earned 15 points')

```

Player earned 5 points  
 Player earned 10 points  
 Player earned 15 points

5-6. Stages of Life: Write an if-elif-else chain that determines a person's stage of life. Set a value for the variable age, and then:

- If the person is less than 2 years old, print a message that the person is a baby.
- If the person is at least 2 years old but less than 4, print a message that the person is a toddler.
- If the person is at least 4 years old but less than 13, print a message that the person is a kid.
- If the person is at least 13 years old but less than 20, print a message that the person is a teenager.
- If the person is at least 20 years old but less than 65, print a message that the person is an adult.
- If the person is age 65 or older, print a message that the person is an elder. if Statements 85

In [ ]:

```

age = 50

if age < 2:
    print('Person is baby')
elif age < 4:
    print('Person is toddler')
elif age < 13:
    print('Person is kid')
elif age < 20:
    print('Person is teenager')
elif age < 65:
    print('Person is adult')
else:
    print('Person is elder')

```

Person is adult

5-7. Favorite Fruit: Make a list of your favorite fruits, and then write a series of independent if statements that check for certain fruits in your list.

- Make a list of your three favorite fruits and call it favorite\_fruits.

- Write five if statements. Each should check whether a certain kind of fruit is in your list. If the fruit is in your list, the if block should print a statement, such as You really like bananas!

```
In [ ]: favorite_fruit = ['squash', 'watermelon', 'melon']

test_fruits = ['squash', 'watermelon', 'melon', 'chips', 'mutton']

fruit = test_fruits[0]
if fruit in favorite_fruit:
    print('You really like ' + fruit)

fruit = test_fruits[1]
if fruit in favorite_fruit:
    print('You really like ' + fruit)

fruit = test_fruits[2]
if fruit in favorite_fruit:
    print('You really like ' + fruit)

fruit = test_fruits[3]
if fruit in favorite_fruit:
    print('You really like ' + fruit)

fruit = test_fruits[4]
if fruit in favorite_fruit:
    print('You really like ' + fruit)
```

```
You really like squash
You really like watermelon
You really like melon
```

5-8. Hello Admin: Make a list of five or more usernames, including the name 'admin'. Imagine you are writing code that will print a greeting to each user after they log in to a website. Loop through the list, and print a greeting to each user. • If the username is 'admin', print a special greeting, such as Hello admin, would you like to see a status report? • Otherwise, print a generic greeting, such as Hello Jaden, thank you for logging in again.

```
In [ ]: usernames = ['admin', 'spandan', 'bhaskar', 'jesus', 'abhishek']

for user in usernames:
    if user == 'admin':
        print('Click yes to see status, ' + user.title() + '?')
    else:
        print('Hello and Welcome to site, ' + user.title())
```

```
Click yes to see status, Admin?
Hello and Welcome to site, Spandan
Hello and Welcome to site, Bhaskar
Hello and Welcome to site, Jesus
Hello and Welcome to site, Abhishek
```

5-9. No Users: Add an if test to hello\_admin.py to make sure the list of users is not empty. • If the list is empty, print the message We need to find some users! • Remove all of the usernames from your list, and make sure the correct message is printed.

```
In [ ]: usernames = []

if usernames ==[]:
    print('We need some Users.')
else:
    for user in usernames:
        if user=='admin':
            print('Click yes to see status, ' + user.title() +'?')
        else:
            print('Hello and Welcome to site, ' + user.title())
```

We need some Users.

5-10. Checking Usernames: Do the following to create a program that simulates how websites ensure that everyone has a unique username. • Make a list of five or more usernames called `current_users`. • Make another list of five usernames called `new_users`. Make sure one or two of the new usernames are also in the `current_users` list. • Loop through the `new_users` list to see if each new username has already been used. If it has, print a message that the person will need to enter a new username. If a username has not been used, print a message saying that the username is available. • Make sure your comparison is case insensitive. If 'John' has been used, 'JOHN' should not be accepted. (To do this, you'll need to make a copy of `current_users` containing the lowercase versions of all existing users.)

```
In [ ]: current_users = ['admin','spandan','bhaskar','jesus','abhishek']

new_users = ['sichitaz','spandan','amartya','jesus','paul']

tempuser = []
for user in current_users:
    lowercase = user.lower()
    tempuser.append(lowercase)

for user in new_users:
    if user.lower() in tempuser:
        print(f'User name {user} unavailabe. Please Enter new user name.')
    else:
        print(f'User name {user} availabe.')
```

User name sichitaz availabe.  
 User name spandan unavailabe. Please Enter new user name.  
 User name amartya availabe.  
 User name jesus unavailabe. Please Enter new user name.  
 User name paul availabe.

5-11. Ordinal Numbers: Ordinal numbers indicate their position in a list, such as 1st or 2nd. Most ordinal numbers end in `th`, except 1, 2, and 3. • Store the numbers 1 through 9 in a list. • Loop through the list. • Use an `if-elif-else` chain inside the loop to print the proper ordinal ending for each number. Your output should read "1st 2nd 3rd 4th 5th 6th 7th 8th 9th", and each result should be on a separate line.

Styling Your if Statements In every example in this chapter, you've seen good styling h

```
In [ ]: numbers = list(range(1,10))

for number in numbers:
    if number == 1:
        print(f'{number}st')
```

```
elif number == 2:  
    print(f'{number}nd')  
elif number == 3:  
    print(f'{number}rd')  
else:  
    print(f'{number}th')
```

1st  
2nd  
3rd  
4th  
5th  
6th  
7th  
8th  
9th