**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  
**Topic 23 - Getting Information from the User and Converting Strings and Numbers**  
**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Using Python’s input() Function**

The input() function in Python lets you get information directly from the user. This is helpful for creating interactive programs where user input drives the program’s actions.

**Example: Checking if a City is Environmentally Clean**

Let’s say we have a list of the five cleanest cities in America. A user wants to check if their city is on that list. To make this work, we need the user to enter their city’s name, which we can collect with input().

python

Copy code

city\_to\_check = input("Enter the name of a city: ")

**How It Works**

* **Prompt Displayed**:  
  The message "Enter the name of a city: " displays on the user’s screen.
* **User Types a City Name**:  
  Python waits for the user to type the name of their city and press Enter.
* **Storing User Input**:  
  The name entered by the user is stored in the variable city\_to\_check.

**Breaking Down the Code**

* **Variable to Store Input**:  
  You need a variable (in this case, city\_to\_check) to hold the user’s input.
* **The input() Function**:  
  input() captures what the user types and assigns it to city\_to\_check.
* **Prompt Message**:  
  The prompt message (inside quotes) is displayed to the user. Here’s how it looks:

python

Copy code

city\_to\_check = input("Enter the name of a city: ")

**Converting User Input: Strings and Numbers**

The input() function always stores data as a **string**. Even if the user types a number, Python treats it as a string by default. This can be a problem when performing calculations.

**Example: Converting User Input to an Integer**

Suppose you ask for a user’s monthly income:

python

Copy code

monthly\_income = input("Enter your monthly income: ")

If the user types **4000**, it’s stored as the string "4000". If you try to calculate annual income by multiplying it by 12, you’ll get an error.

**Converting a String to an Integer or a Float**

To perform calculations, convert the input to an integer or float.

1. **Convert to Integer (int)**:  
   Use int() to change a string to an integer:

python

Copy code

monthly\_income\_as\_integer = int(monthly\_income)

1. **Convert to Float (float)**:  
   Use float() to get a decimal number:

python

Copy code

monthly\_income\_as\_float = float(monthly\_income)

**Converting Numbers Back to Strings**

Sometimes, you’ll need to convert a number back to a string to display it with text. For example, suppose you have the minimum wage stored as an integer:

python

Copy code

min\_wage = 15

If you try to print it with text:

python

Copy code

print("The minimum wage in your state is $" + min\_wage)

This will cause an error, as you can’t combine a string and an integer directly.

**Solution: Convert to String (str)**

Convert the integer to a string with str():

python

Copy code

min\_wage = str(min\_wage)

print("The minimum wage in your state is $" + min\_wage)

**Summary of Conversions**

1. **Convert to Integer**:

python

Copy code

int(variable\_name)

1. **Convert to Float**:

python

Copy code

float(variable\_name)

1. **Convert to String**:

python

Copy code

str(variable\_name)