

Introduction

Python, a high-level scripting language, is renowned for its simplicity and efficiency. In the realm of cybersecurity, Python is a valuable asset, facilitating the creation of tools for hacking, defence, and analysis. This report provides a comprehensive overview of Python's basic concepts, aiming to equip learners with the skills to craft basic scripts for various cybersecurity applications.

<https://tryhackme.com/p/Damiano254>

Task 1: Introduction to Python

Python is celebrated for its clear syntax and powerful capabilities. Essential concepts covered include:

- **Variables:** Essential for storing and manipulating data.
- **Loops:** Crucial for executing repetitive tasks without manual input, thereby saving time and reducing errors.
- **Functions:** Blocks of code designed for a specific task, reusable throughout the program, enhancing code organization and efficiency.
- **Data Structures:** Methods for organizing data, which is fundamental for managing and processing large amounts of information effectively.
- **If statements:** Control structures that allow for conditional execution of code segments, vital for decision-making processes in programming.
- **Files:** Reading and writing files are fundamental operations, especially in cybersecurity, where data handling is a routine task.

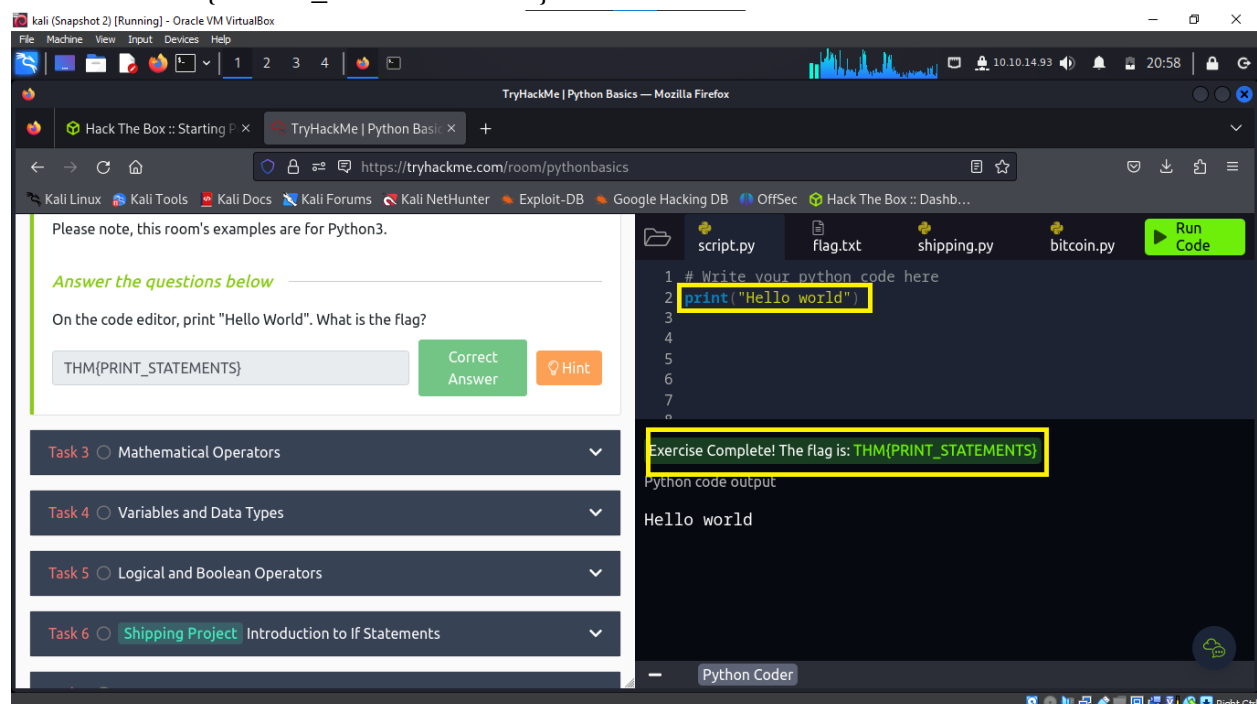
Exercise

Practice writing Python scripts to reinforce learning and understanding of these concepts.

Question

What output does printing "Hello World" in Python generate?

Answer: THM{PRINT_STATEMENTS}



Task 2: Hello World

The "Hello World" program is the starting point for learning any programming language. It introduces basic syntax, the concept of comments (non-executable explanations in the code), and the print function, a fundamental method for outputting data to the screen.

Question

Demonstrate basic arithmetic operations in Python.

Answers:

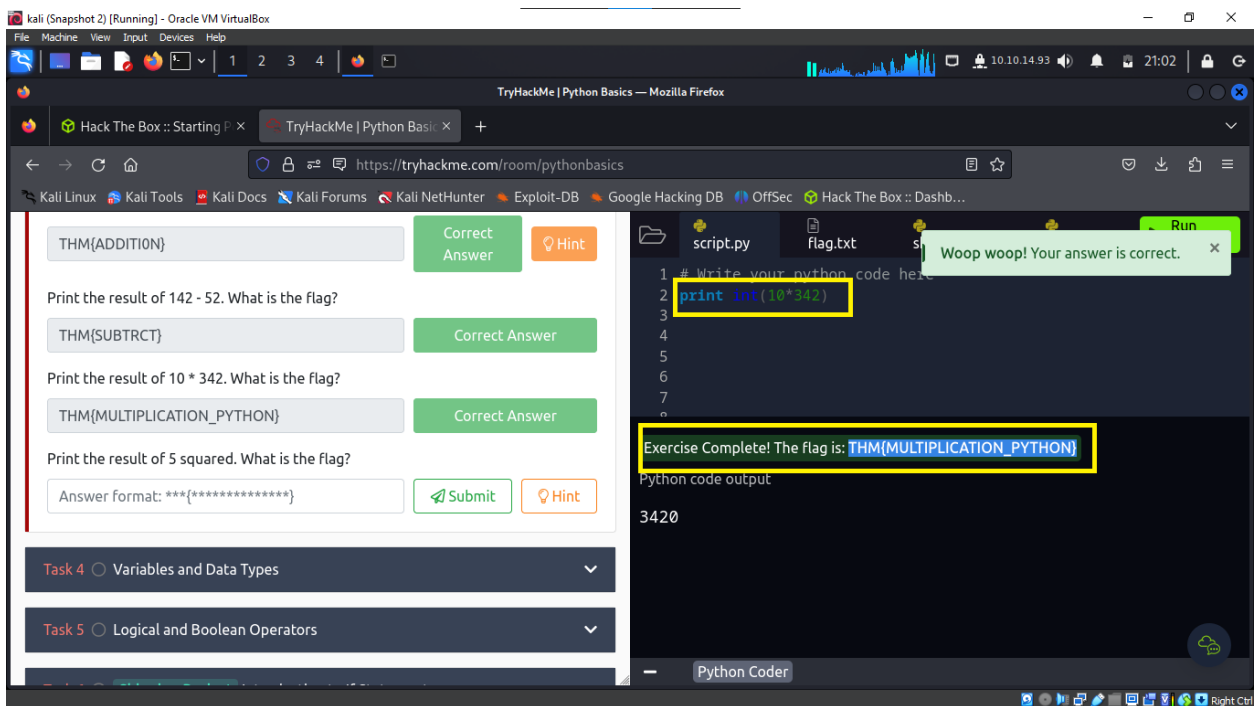
- For addition: THM{ADDITION}

The screenshot shows the TryHackMe Python Basics exercise interface. On the left, a list of comparison operators is displayed: Not Equal to (`!=`), Greater than or equal to (`>=`), and Less than or equal to (`<=`). Below this, a section titled "Answer the questions below" contains three questions: "In the code editor, print the result of 21 + 43. What is the flag?", "Print the result of 142 - 52. What is the flag?", and "Print the result of 10 * 342. What is the flag?". The first question is selected, and the answer field contains "THM{ADDITION}". A green "Correct Answer" button is visible. On the right, a code editor shows a Python script with the line `print(int(21+43))` highlighted. A green notification bubble says "Woop woop! Your answer is correct." Below the code editor, a message states "Exercise Complete! The flag is: THM{ADDITION}" and the Python code output is shown as "64".

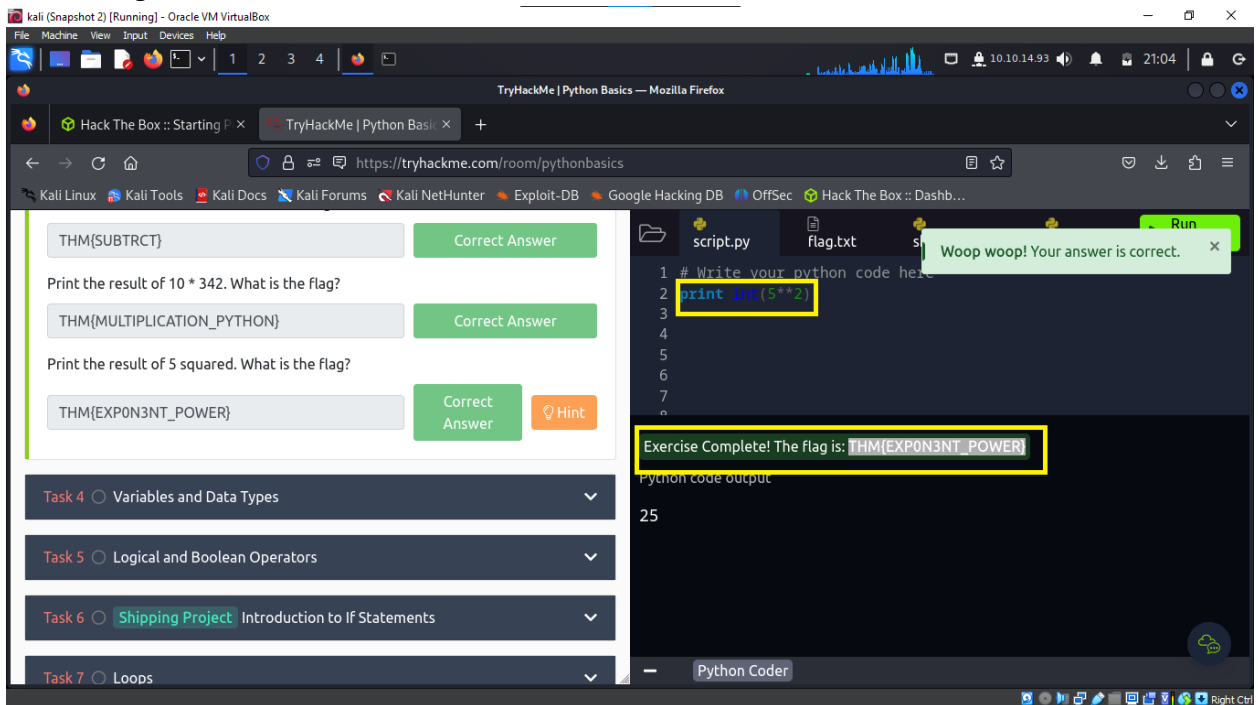
- For subtraction: THM{SUBTRACT}

The screenshot shows the TryHackMe Python Basics exercise interface. On the left, a list of comparison operators is displayed: Not Equal to (`!=`), Greater than or equal to (`>=`), and Less than or equal to (`<=`). Below this, a section titled "Answer the questions below" contains three questions: "In the code editor, print the result of 21 + 43. What is the flag?", "Print the result of 142 - 52. What is the flag?", and "Print the result of 10 * 342. What is the flag?". The second question is selected, and the answer field contains "THM{SUBTRACT}". A green "Correct Answer" button is visible. On the right, a code editor shows a Python script with the line `print(int(142-52))` highlighted. A green notification bubble says "Woop woop! Your answer is correct." Below the code editor, a message states "Exercise Complete! The flag is: THM{SUBTRACT}" and the Python code output is shown as "90".

- For multiplication: THM{MULTIPLICATION_PYTHON}



- For exponentiation: THM{EXP0N3NT_POWER}



Task 3: Mathematical Operators

Python's mathematical capabilities are akin to a calculator. It supports various operations crucial for numerical data processing. This feature is particularly useful in cybersecurity for tasks such as encryption, decryption, and algorithmic analysis.

Question

Implement simple arithmetic operations in Python.

Answer: Given in the task details.

Task 4: Variables and Data Types

Understanding variables and data types is fundamental in Python. Variables act as storage containers for data, which can be of various types like strings (text), integers (whole numbers), floats (numbers with decimals), booleans (true/false), and lists (ordered sequences of items).

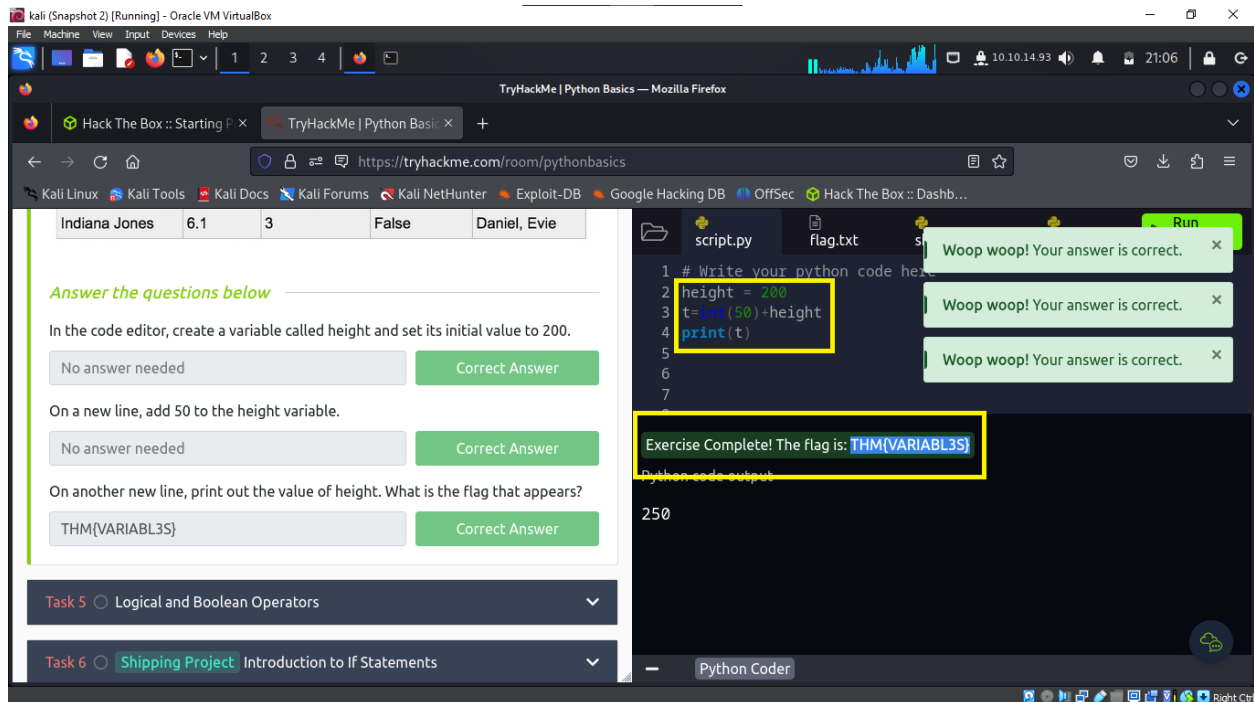
Exercise

Experiment with variables and observe how data manipulation and storage work in Python.

Question

Modify and print a variable's value in Python.

Answer: THM{VARIABLE3S}



Task 5: Logical and Boolean Operators

Logical and boolean operators are critical in Python for making comparisons and logical decisions. They form the backbone of control flow in programming, enabling the execution of code based on specific conditions.

Question

Apply logical and boolean operators in Python.

Answer: Not specified in the notes.

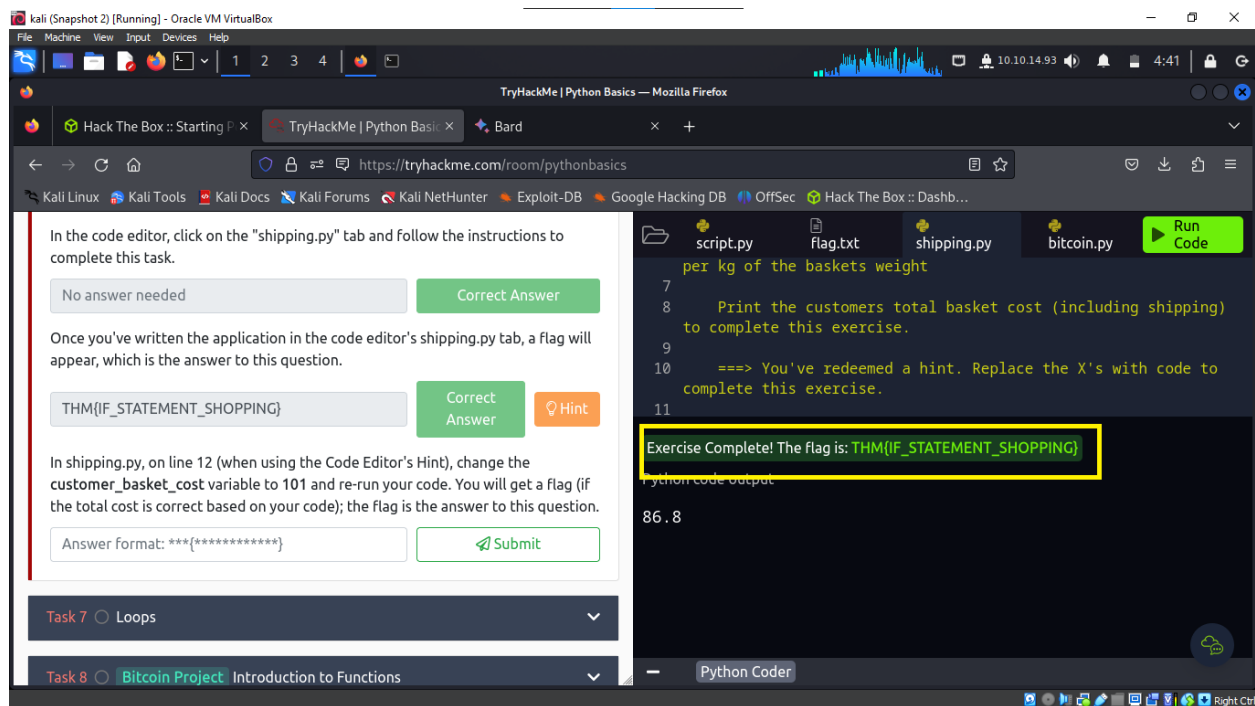
Task 6: Shipping Project and If Statements

If statements are the building blocks of decision-making in Python. They allow the execution of certain code segments based on specific conditions, which is a fundamental concept in developing applications that require dynamic responses to user input or changing data.

Question

Create a Python application to calculate and output shipping costs.

Answer: THM{IF_STATEMENT_SHOPPING}



Task 7: Loops

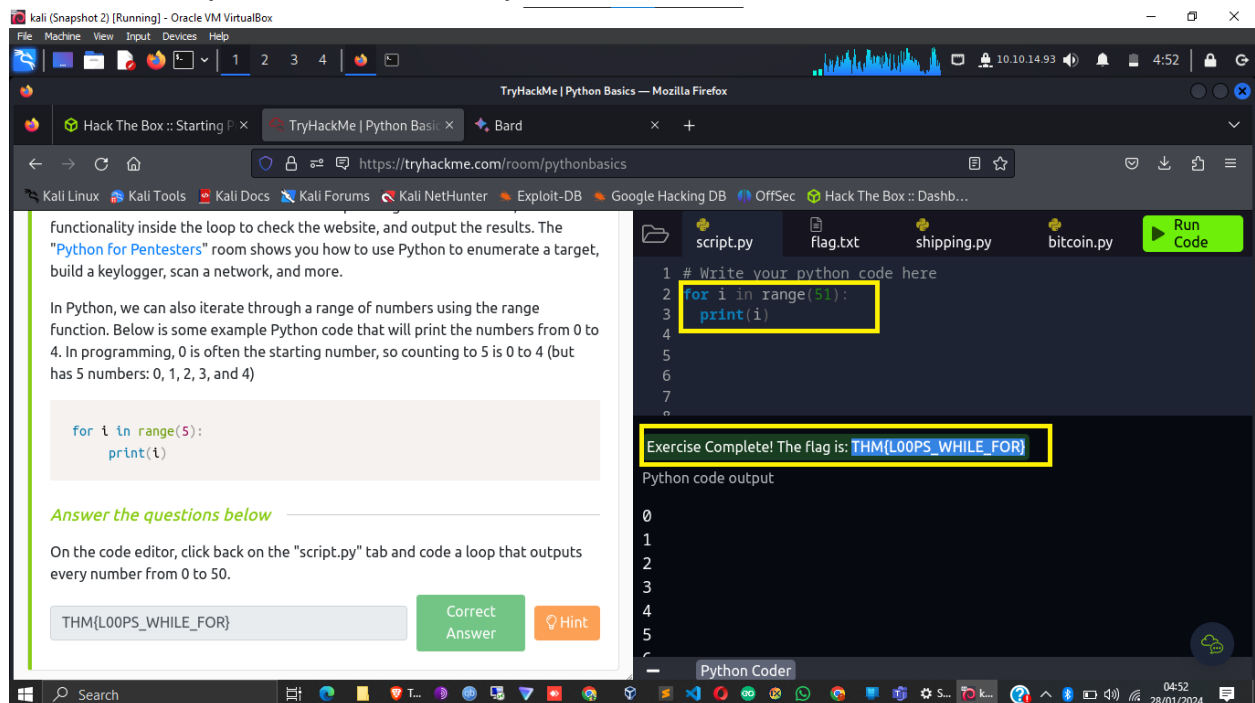
Loops are essential for automating repetitive tasks in Python. The two main types of loops are:

- **While Loops:** These continue to execute as long as a certain condition is true, useful for tasks where the number of iterations is not known in advance.
- **For Loops:** Ideal for iterating over a sequence (like a list) or a range of numbers, thereby streamlining operations that involve data structures.

Question

Develop a loop that prints a range of numbers.

Answer: `THM{L00PS_WHILE_FOR}`



Task 8: Bitcoin Project and Introduction to Functions

Functions in Python are blocks of code designed to perform a specific task and can be reused throughout the program. They enhance the modularity and efficiency of the code, which is crucial in complex programming projects, including those in cybersecurity.

Question

Write and utilize a function in Python to calculate the USD value of Bitcoin and determine if it falls below a certain threshold.

Answer: THM{BITCOIN_INVESTOR}

The screenshot shows a web browser window with the URL `https://tryhackme.com/room/pythonbasics`. The page displays a task description: "with two parameters: `bitcoin_amount`, the amount of Bitcoin you own, and `bitcoin_value_usd`, the value of bitcoin in USD. The function should return `usd_value`, which is your bitcoin value in USD (to calculate this, in the function, you times `bitcoin_amount` variable by `bitcoin_value_usd` variable and return the value). The start of the function should look like this:

```
def bitcoinToUSD(bitcoin_amount, bitcoin_value_usd):
```

Below the description, there is a text input field containing the answer `THM{BITCOIN_INVESTOR}`. A green "Correct Answer" button is visible. The right side of the screen shows a code editor with the following Python code:

```
16 script.py
17 def bitcoinToUSD(bitcoin_amount, bitcoin_value_usd):
18     return bitcoin_amount * bitcoin_value_usd
19
20 money_usd = bitcoinToUSD(investment_in_bitcoin, bitcoin_to_usd)
21
22 if money_usd < 30000:
23     print("alert")
```

A yellow box highlights the `if` statement. Below the code editor, a message states: "Exercise Complete! The flag is: THM{BITCOIN_INVESTOR}".

Task 9: Files

File handling is a critical aspect of Python, particularly in cybersecurity, where reading from and writing to files is a common practice. Python's file handling capabilities enable efficient data processing and storage, which is essential for tasks like logging, data extraction, and script output management.

Question

Read the contents of a file using Python.

Answer: THM{FILE_READ}

The screenshot shows a web browser window with the URL `https://tryhackme.com/room/pythonbasics`. The page displays a task description: "Notice we use the `close()` method after writing to a file; this closes the file so no more writing to the file (within the program) can occur."

Below the description, there is a text input field containing the answer `THM{FILE_READ}`. A green "Correct Answer" button is visible. The right side of the screen shows a code editor with the following Python code:

```
1 f = open("flag.txt", "r")
2 print(f.read())
```

A yellow box highlights the `print(f.read())` line. Below the code editor, a message states: "Woop woop! Your answer is correct."

Task 10: Imports

Python's ability to import libraries expands its functionality significantly. Libraries are collections of pre-written functions and modules that provide additional capabilities, such as network packet manipulation or HTTP request handling, enhancing Python's utility in cybersecurity contexts.

Exercise

Explore and utilize Python libraries to understand their practical applications in scripting and cybersecurity.

Conclusion

Python is an invaluable tool in the cybersecurity field, offering a wide range of functionalities. Mastery of Python basics, including variables, loops, functions, and file handling, is crucial for anyone looking to leverage Python in cybersecurity. The language's simplicity, coupled with its powerful capabilities, makes it a preferred choice for both beginners and experienced professionals in the field.

The screenshot shows a web browser window with the URL `tryhackme.com/p/Damiano254`. The browser's address bar and tabs are visible at the top. The main content area displays the user profile for **Damiano254** with a rank of 374009, 7 rooms completed, level 3, and 0 badges. Below the profile stats, there are tabs for 'Rooms Complete', 'Badges', 'Created Rooms', 'Yearly Activity', and 'Tickets'. The 'Rooms Complete' tab is active, showing a grid of room cards. The 'Python Basics' room card is highlighted with a blue border. It features a Python logo and the text 'Python Basics' and 'Using a web-based code editor, learn the basics of...'. Other visible room cards include 'Web Application...', 'Intro to Offensiv...', 'Intro to Digital...', 'Red Team Recon', 'MITRE', and 'Simple CTF'.

Rank	Rooms Complete	Level	Badges
374009	7	3	0

Damiano254 [0x3]

Get Profile Badge ID | Share Room Badges

Rooms Complete | Badges | Created Rooms | Yearly Activity | Tickets

- Web Application... Learn about web applications and explore...
- Intro to Offensiv... Hack your first website (legally in a safe...)
- Intro to Digital... Learn about digital forensics and related...
- Red Team Recon Learn how to use DNS, advanced searching, Reco...
- Python Basics** Using a web-based code editor, learn the basics of...
- MITRE This room will discuss the various resources MITRE h...
- Simple CTF Beginner level ctf 010

<https://tryhackme.com/room/redteamrecon>