

We can know which module is running directly and which is module is running using import

So we can distinguish between module running directly and one that is importing

So both behaves differently

The screenshot shows two Jupyter notebooks side-by-side. The left notebook, titled 'm1.py', contains a single line of code: `print("M1 Module %s",(__name__))`. The right notebook, titled 'm2.py', contains three lines: `import m1`, `2`, and `print("M2 Module %s",(__name__))`. A Windows Command Prompt window is overlaid in the center, showing the execution of these scripts. It displays the directory path `C:\Users\ARYAN GULATI\Desktop\Python_Code\2.)PYTHON_BASICS\PYTHON Basics` and the results of running `python m1.py` and `python m2.py`. The output for `m1.py` is `M1 Module %s __main__`, and for `m2.py` it is `M1 Module %s m1` followed by `M2 Module %s __main__`.

The screenshot shows two Jupyter notebooks side-by-side. The left notebook, titled 'm2.py', contains three lines: `import m1`, `2`, and `print("M2 Module %s",(__name__))`. The right notebook, titled 'm1.py', contains a conditional block: `if __name__ == "__main__":`, `# what to do when this module run directly`, `print("M1 Module %s",(__name__))`, `6`, `else:`, `#specify what to do when this module is imported`, and `print("I am in M1,else block")`. A Windows Command Prompt window is overlaid in the center, showing the execution of these scripts. It displays the directory path `C:\Users\ARYAN GULATI\Desktop\Python_Code\2.)PYTHON_BASICS\PYTHON Basics` and the results of running `python m1.py` and `python m2.py`. The output for `m1.py` is `M1 Module %s __main__`, and for `m2.py` it is `M1 Module %s m1` followed by `M2 Module %s __main__`. The output for `m1.py` when imported is `I am in M1,else block`.