ERRORS OCCURRED DURING THE CODING PHASE

Error-1: ModuleNotFoundError: No module named 'functions'

Error Screenshot:

```
---------------------------------------------------------------------------
ModuleNotFoundError                       Traceback (most recent call last)
Cell In[1], line 3
      1 from shiny import App, ui
      2 from shiny.express import render, input
----> 3 from functions import umap_process, extractzip
      4 import os
      6 def app_ui():
      7     # (main.py içeriği)

ModuleNotFoundError: No module named 'functions'
```

Why the Error Might Have Occurred?:

1. The file `functions.py` (containing the required functions) is missing, in the wrong location, or has a typo in its name.
2. The directory with `functions.py` isn't in Python's search path.
3. There is an incorrect project structure or import statement.

Why Did It Really Happen?:

- I forgot to put the functions.py file in the folder. (I know stupid mistake :/)

How to Solve The Issue?:

- As easy as it sounds, I've carried the functions.py to the folder.

Error-2: IndentationError: expected an indented block after function definition on line 33

Error Screenshot:

```
  Cell In[18], line 37
    app = App(app_ui(), server)  # Creating the Shiny application
    ^
IndentationError: expected an indented block after function definition on line 33
```

Why the Error Might Have Occurred?:

1. **Missing Indentation:** In Python, code blocks within functions, loops, and conditional statements are defined by indentation. This error means that after defining a function (presumably on line 33), the code that should be inside that function is not indented.
2. **Incorrect Indentation Level:** Even if there is indentation, it might not be consistent or at the correct level (usually four spaces or one tab).
3. **Empty Function:** It might have been defined as a function with no code inside it. Python requires at least one indented statement within a function.

Why Did It Really Happen?:

- There were too many spaces.

How to Solve The Issue?:

- Make sure all lines within the function are indented at the same level. Inconsistent indentation can also cause this error.

Error-3: RuntimeError: asyncio.run() cannot be called from a running event loop

Error Screenshot:

```
---------------------------------------------------------------------------
RuntimeError                              Traceback (most recent call last)
Cell In[3], line 107
    105 # Run the app
    106 if __name__ == "__main__":
--> 107     app.run()

File /opt/anaconda3/lib/python3.12/site-packages/shiny/_app.py:271, in App.run(self, **kwargs)
    261 """
    262 Run the app.
    263
   (...)
    267     Keyword arguments passed to :func:`~shiny.run_app`.
    268 """
    269 from ._main import run_app
--> 271 run_app(self, **kwargs)

File /opt/anaconda3/lib/python3.12/site-packages/shiny/_main.py:367, in run_app(app, host, port, autoreload_port, reload, reload_dirs, reload_includes, reload_excludes, ws_max_size, lo
g_level, app_dir, factory, launch_browser, dev_mode, **kwargs)
    363     setup_launch_browser(log_config)
    365 maybe_setup_rsw_proxying(log_config)
--> 367 uvicorn.run(  # pyright: ignore[reportUnknownMemberType]
    368     app,
    369     host=host,
    370     port=port,
    371     ws_max_size=ws_max_size,
    372     log_level=log_level,
    373     log_config=log_config,
    374     app_dir=app_dir,
    375     factory=factory,
    376     lifespan="on",
    377     # Don't allow shiny to use uvloop!
    378     # https://github.com/posit-dev/py-shiny/issues/1373
    379     loop="asyncio",
    380     **reload_args,  # pyright: ignore[reportArgumentType]
    381     **kwargs,
    382 )

File /opt/anaconda3/lib/python3.12/site-packages/uvicorn/main.py:579, in run(app, host, port, uds, fd, loop, http, ws, ws_max_size, ws_max_queue, ws_ping_interval, ws_ping_timeout, ws_
per_message_deflate, lifespan, interface, reload, reload_dirs, reload_includes, reload_excludes, reload_delay, workers, env_file, log_config, log_level, access_log, proxy_headers, serv
er_header, date_header, forwarded_allow_ips, root_path, limit_concurrency, backlog, limit_max_requests, timeout_keep_alive, timeout_graceful_shutdown, ssl_keyfile, ssl_certfile, ssl_ke
yfile_password, ssl_version, ssl_cert_reqs, ssl_ca_certs, ssl_ciphers, headers, use_colors, app_dir, factory, h11_max_incomplete_event_size)
    577         Multiprocess(config, target=server.run, sockets=[sock]).run()
    578     else:
--> 579         server.run()
    580 except KeyboardInterrupt:
    581     pass  # pragma: full coverage

File /opt/anaconda3/lib/python3.12/site-packages/uvicorn/server.py:66, in Server.run(self, sockets)
     64 def run(self, sockets: list[socket.socket] | None = None) -> None:
     65     self.config.setup_event_loop()
---> 66     return asyncio.run(self.serve(sockets=sockets))

File /opt/anaconda3/lib/python3.12/asyncio/runners.py:190, in run(main, debug, loop_factory)
    161 """Execute the coroutine and return the result.
    162
    163 This function runs the passed coroutine, taking care of
   (...)
    186     asyncio.run(main())
    187 """
    188 if events._get_running_loop() is not None:
    189     # fail fast with short traceback
--> 190     raise RuntimeError(
    191         "asyncio.run() cannot be called from a running event loop")
    193 with Runner(debug=debug, loop_factory=loop_factory) as runner:
    194     return runner.run(main)
```

Why the Error Might Have Occurred?:

- **Nested `asyncio.run()` Calls:** The primary reason for this error is that you're trying to call `asyncio.run()` from within an already running asyncio event loop. `asyncio.run()` is designed to create and manage the top-level event loop, and it cannot be used when an event loop is already active.
- **Shiny and Uvicorn:** This error specifically arises within the context of your Shiny application, which uses Uvicorn as its ASGI server. Uvicorn itself runs within an asyncio event loop, and when `app.run()` is called, it triggers another attempt to create a new event loop, leading to the error.
- **Incorrect Application Structure:** It may be that the application is structured in a way that is causing the shiny app to attempt to create an event loop when it should not.

Why Did It Really Happen?:

- I have tried to find why it might have “really” happened, my best answer is that the Jupyter Notebook isn’t compatible with Shiny App, so when being tried to run it gives a major error.

How to Solve The Issue?:

- In theory:
    - **Avoid Nested `asyncio.run()`:** The core solution is to ensure that `asyncio.run()` is only called once at the top level of your application. In this case, you should not be calling `asyncio.run()` from inside of the shiny application, as the shiny application is already running an event loop.
    - **Use `await` Instead:** If you need to execute asynchronous code within your Shiny application, use the `await` keyword to schedule coroutines within the existing event loop.
    - **Shiny's `app.run()`:** Shiny's `app.run()` function handles the event loop for you. You should not attempt to manage the event loop manually when using Shiny's `app.run()`.
    - **Review Your Application's `if __name__ == "__main__":` Block:** The error occurs in line 107, within the `if __name__ == "__main__":` block. Ensure that you're only calling `app.run()` here and not attempting to run any other asynchronous code using `asyncio.run()`.
    - **Ensure Proper Shiny Usage:** Make sure that you are using shiny the way it is intended. Shiny manages the event loop. Therefore, you should not be attempting to manage it yourself.
    - **Update Packages:** Ensure that you are using the latest versions of Shiny and Uvicorn. Sometimes, bugs related to event loop management are fixed in newer releases.
- In reality:
    - I have tried all of these methods, either I didn't do something properly or the error is bigger/not related to the ways of solution above. I still have this error in occurred position.

Error-4: AttributeError: 'App' object has no attribute 'serve'

Error Screenshot:

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[1], line 95
     93 # Ensure compatibility with Jupyter Notebook
     94 if __name__ == "__main__":
---> 95     asyncio.run(app.serve())

AttributeError: 'App' object has no attribute 'serve'
```

Why the Error Might Have Occurred?:

- **Incorrect Method Call:** You're attempting to call the `serve()` method on an `App` object, but the `App` object from the `shiny` library (which is what your code indicates you're using) does not have a method named `serve()`.
- **Version Mismatch or API Change:** It's possible that you're using an older version of the `shiny` library where `serve()` was not a valid method, or that the API has changed, and `serve()` has been replaced or removed.
- **Typo or Misunderstanding:** There might be a typo in the method name, or you might be mistakenly thinking that the `App` object has a `serve()` method.
- **Incorrect Usage:** You are trying to run the shiny application in a way that is not intended.

Why Did It Really Happen?:

- In error-3, I have had an issue with asyncio, since it wasn't (still hasn't been) solved, this might have naturally occurred. I have tried to find why it might have "really" happened, my best answer is that the Jupyter Notebook isn't compatible with Shiny App, so when being tried to run it gives a major error.

How to Solve The Issue?:

- In theory:
  - **Use `app.run()`:** The correct method to run a Shiny application is `app.run()`. Replace `app.serve()` with `app.run()`.
  - **Verify Shiny Version:** Ensure that you are using the latest version of the `shiny` library. If you are using an older version, consider updating it.
  - **Check Shiny Documentation:** Refer to the official Shiny documentation for the correct way to run your application.
  - **Remove `asyncio.run()`:** Shiny applications run their own event loop. Therefore you do not need to wrap the `app.run()` command in an `asyncio.run()` command.
  - **Correct Usage:** The correct way to run the application in the `if __name__ == "__main__":` block is `app.run()`.

- In reality:
  - I have tried all of these methods, either I didn't do something properly or the error is bigger/not related to the ways of solution above. I still have this error in the position.

Error-5: OSError: [Errno 48] error while attempting to bind on address ('127.0.0.1', 8000): address already in use

Error Screenshot:

Why the Error Might Have Occurred?:

- **Port Conflict:** This error means that the port you're trying to use (port 8000 in this case) is already being used by another application on your system.
- **Previous Instance Running:** A previous instance of your Shiny application (or another application) might still be running and holding onto that port.
- **Zombie Process:** A process that was supposed to terminate might have become a "zombie" process, still holding the port without actively doing anything.
- **Operating System Behavior:** Sometimes, the operating system might take a short time to release a port after a process has terminated.

Why Did It Really Happen?:

- I don’t really know why it “actually” happened, but my best guess is Port Conflict, since I’ve tried dozens of different coding versions and this one is still open somewhere.

How to Solve The Issue?:

Identify and Terminate Conflicting Process:

- Use your operating system's tools to identify the process that's using port 8000.
- On Linux/macOS, you can use `lsof -i :8000` or `netstat -tuln | grep 8000`.
- On Windows, you can use `netstat -ano | findstr :8000` and then use `taskkill /F /PID [PID]` to terminate the process.

Change the Port:

- If you can't terminate the conflicting process, or if you want to run multiple instances of your application, change the port number that your Shiny application is using.
- This is typically done in the app.run() command.

Wait and Retry:

- If you recently terminated a process that was using port 8000, wait a few seconds and try running your application again.

Restart Your Computer:

- As a last resort, restarting your computer can clear out any lingering processes that might be holding onto the port.

What I’ve Done?

- I’ve downloaded the app.py on my computer and used “cd” to call my desktop on the terminal, after that i’ve written “shiny run app.py –port 8001 and made it work.