Install Bandit – static analyzer tool for python

Command: <mark>pip install bandit</mark>

Syntax to rum bandit on a python code

<mark>bandit -r project_name -f html -o outputfile</mark>

-r → perform security analysis recursively

Project_name → name of the file

-f → type of output file, in this case it is html

-o → output file

Outputfile → name of output file

Lets perform for the folder name as 'try'

Bandit allows you to output results in JSON format, providing more structured information. You can use the -f or --format option to specify the output format. For example:
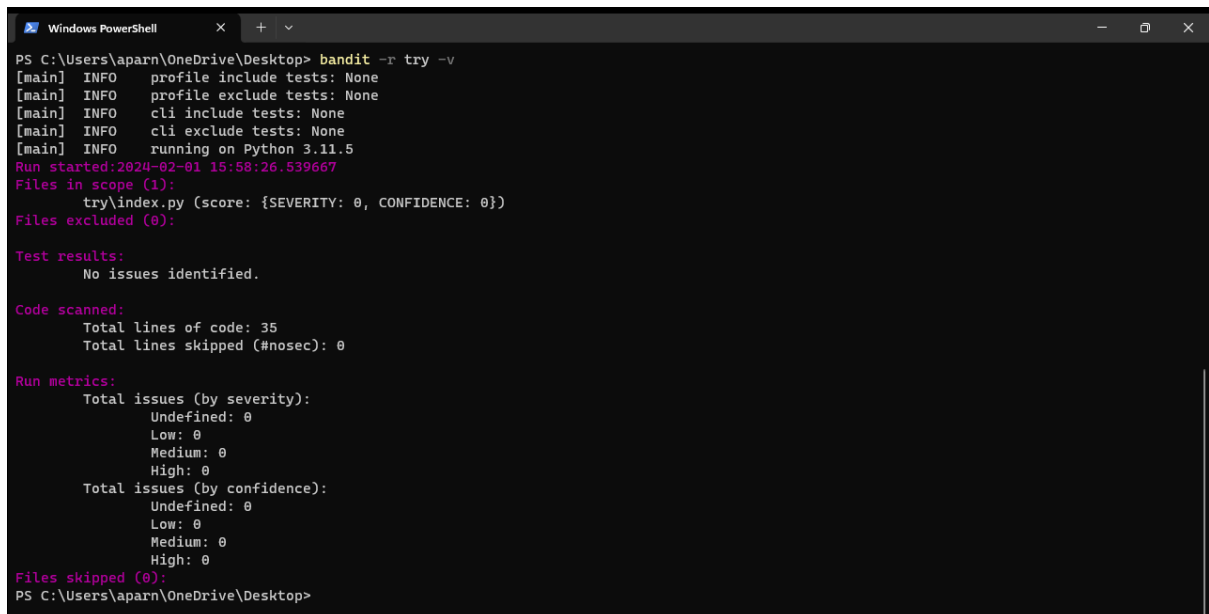
bandit -r try -f json -o bandit_results.json

```
PS C:\Users\aparn\OneDrive\Desktop> bandit -r try -f json -o bandit_results.json
[main]  INFO     profile include tests: None
[main]  INFO     profile exclude tests: None
[main]  INFO     cli include tests: None
[main]  INFO     cli exclude tests: None
[json]  INFO     JSON output written to file: bandit_results.json
PS C:\Users\aparn\OneDrive\Desktop>
```

Here is the result:-

```json
{
  "errors": [],
  "generated_at": "2024-02-01T15:54:44Z",
  "metrics": {
    "_totals": {
      "CONFIDENCE.HIGH": 0,
      "CONFIDENCE.LOW": 0,
      "CONFIDENCE.MEDIUM": 0,
      "CONFIDENCE.UNDEFINED": 0,
      "SEVERITY.HIGH": 0,
      "SEVERITY.LOW": 0,
      "SEVERITY.MEDIUM": 0,
      "SEVERITY.UNDEFINED": 0,
      "loc": 35,
      "nosec": 0,
      "skipped_tests": 0
    },
    "try\\index.py": {
      "CONFIDENCE.HIGH": 0,
      "CONFIDENCE.LOW": 0,
      "CONFIDENCE.MEDIUM": 0,
      "CONFIDENCE.UNDEFINED": 0,
      "SEVERITY.HIGH": 0,
      "SEVERITY.LOW": 0,
      "SEVERITY.MEDIUM": 0,
      "SEVERITY.UNDEFINED": 0,
      "loc": 35,
      "nosec": 0,
      "skipped_tests": 0
    }
  },
  "results": []
}
```

Verbose Output: The -v or --verbose option can be used to increase verbosity, providing more information about the analysis. For example:

bandit -r try -v