

# Debugging Code

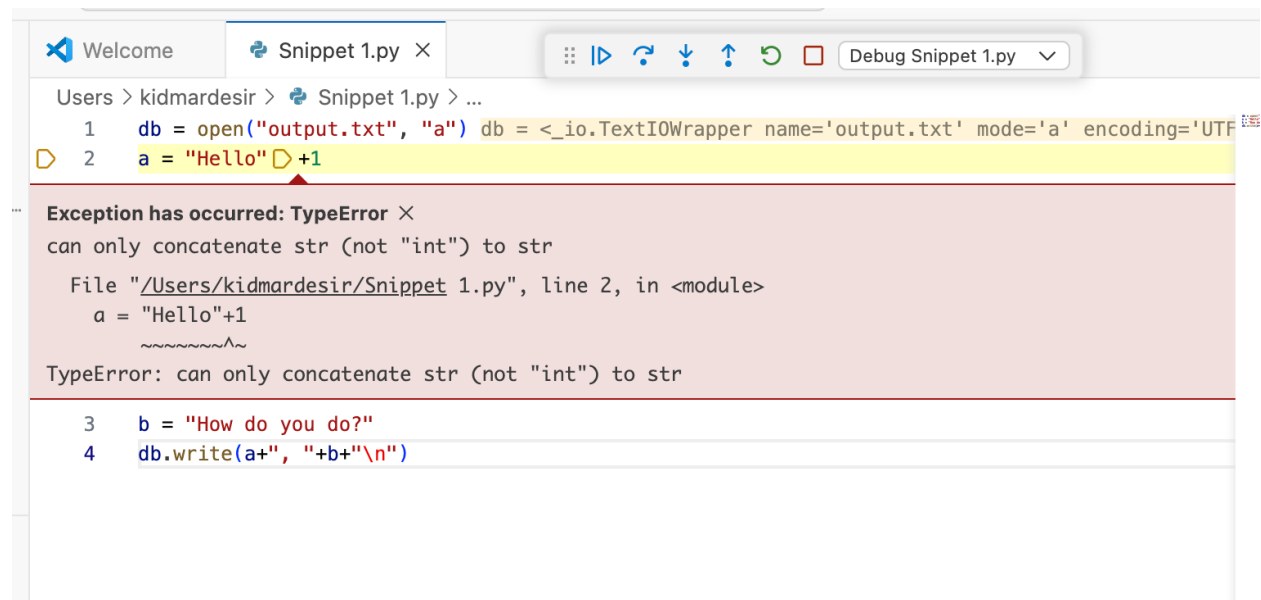
## Report by Kidmar Desir

---

### Code Snippet 1:

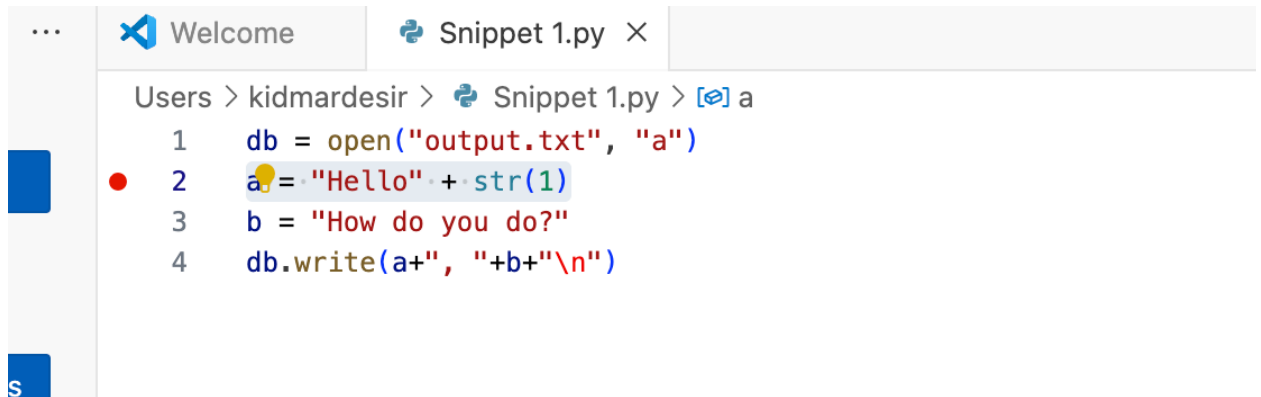
```
db = open("output.txt", "a")
a = "Hello"+1
b = "How do you do?"
db.write(a+", "+b+"\n")
```

1. When I debugged the code, it attempts to concatenate a string("Hello") with an integer(1), which is not allowed in Python. This generated a `TypeError`.



2. The file `output.txt` is opened but not closed properly. It is better to close the file by using a `<<with>>` statement.

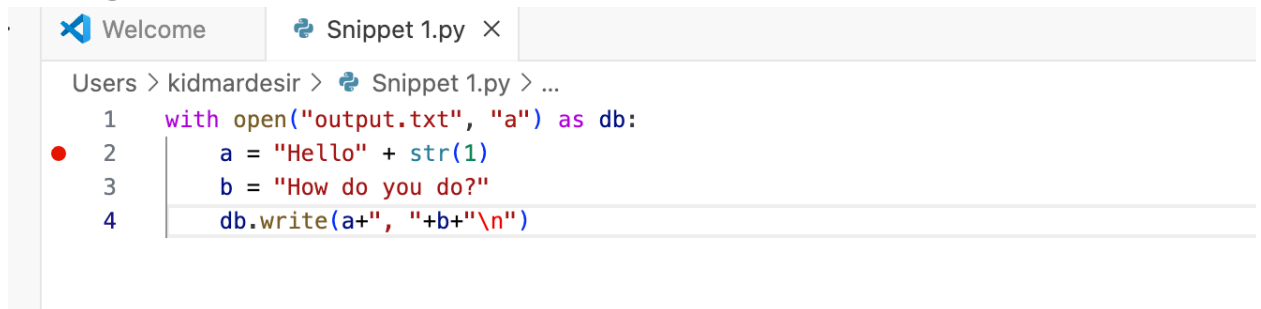
3. To fix the errors I converted the integer <<1>> to a string before concatenation



The screenshot shows a code editor with a tab titled 'Snippet 1.py'. The file path is 'Users > kidmardesir > Snippet 1.py'. The code contains four lines: 1. `db = open("output.txt", "a")`, 2. `a = "Hello" + str(1)` (with a red dot indicating an error), 3. `b = "How do you do?"`, and 4. `db.write(a+, "+b+"\n")`. The error is a syntax error on line 2, where the integer 1 is not converted to a string before concatenation.

```
1 db = open("output.txt", "a")
2 a = "Hello" + str(1)
3 b = "How do you do?"
4 db.write(a+, "+b+"\n")
```

4. Use a <<with>> statement to ensure the file is properly closed after writing.



The screenshot shows a code editor with a tab titled 'Snippet 1.py'. The file path is 'Users > kidmardesir > Snippet 1.py > ...'. The code contains four lines: 1. `with open("output.txt", "a") as db:`, 2. `a = "Hello" + str(1)`, 3. `b = "How do you do?"`, and 4. `db.write(a+, "+b+"\n")`. The code is syntactically correct and uses a `with` statement to ensure the file is properly closed after writing.

```
1 with open("output.txt", "a") as db:
2     a = "Hello" + str(1)
3     b = "How do you do?"
4     db.write(a+, "+b+"\n")
```