Exception handing in python

Exception handing

Exception handing is the process of responding to unwanted or unexpected events when a computer program runs. Exception handing deals with these events to avoid the program or system crashing, and without the process, exceptions would disrupt the normal operation of a program.

Exception in python

Python has many built in exception that are raised when your program encounters an error (something in the program goes wrong).

When these exception occur, the Python interpreter stops the current process and passes it to the calling process until it is handled. If not handled, the program the program will crash.

Python try except

Try except blocks are used in python to handle errors and exceptions. The code in try block runs when there is no error. If the try block catches the error, then the except block is executed.

Example:

```
try:
    num=int(input("Enter the number"))
    a=[6,3]
    print(a[num])
except ValueError:
    print("please Number enter an integer")
except IndexError:
    print("Index error")
```

Source Code

```
# # a =int(input("Enter the number:"))
```

```python
# print(f"Multipalication table of {a} is :")
# try:
#       for i in range(1,11):
#           print(f"{{int(a) X {i} {i}={int(a)*i}}}")
# except:
#     print("Invalid Error Occur")

# print("Some important line of Code")
# print("End of program")


# ValueError
try:
    num=int(input("Enter the number"))
    a=[6,3]
    print(a[num])
except ValueError:
    print("please Number enter an integer")
except IndexError:
    print("Index error")
```

Thank You