

Database vs program



Employee ID	Name	Dept	Salary
100	John	HR	25500
101	Mary	R&D	44500
102	Bill	R&D	43000
103	Tom	Sales	40000

SELECT * FROM employees
WHERE Salary > 42000;

```
SELECT * FROM employees
WHERE Salary < 30000;</pre>
```



Variables

Variables are named areas in the computer's memory that store values.



Variables



Variables are named areas in the computer's memory that store values.



<u>IF</u> statements



```
age = 17

if (age > 17):
    print("OK")

print("Finished")
```

Finished

```
age = 17

if (age > 17):
    print("OK")
elif (age < 18):
    print("NOK")
print("Finished")

NOK
Finished</pre>
```

```
if (temp > 37):
    print("Hot")
elif (temp < 37):
    print("Cold")
else:
    print("OK")
print("Finished")

OK
Finished</pre>
```



<u>input</u>



```
email = name + "@gmit.ie"
```

print(email)

Tom@gmit.ie

```
salary = input("Enter salary")
```

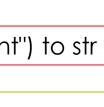
salary = int(salary)

salary = salary + 100

print(salary)

TypeError: can only concatenate str (not "int") to str







WHILE statement

```
i = 1
while (i <= 5):
   print(i)
   i += 1
   # i = i + 1
```

```
answer = "5";
while True:
    guess = input("Pick a number between 1 & 10")
    if (guess == answer):
        print("Correct!")
        break;
print("End")
```



<u>ARRAYS</u>





append()



```
myArr = ["Jan", "Feb", "March", "April"]

myArr.append("May")

myArr = myArr.append("May")

print(myArr)

['Jan', 'Feb', 'March', 'April', 'May']
```



FOR statement



```
names = ["Tom", "John", "Mary", "Bob"]
for name in names:
    print (name+"@gmit.ie")
```

Tom@gmit.ie John@gmit.ie Mary@gmit.ie Bob@gmit.ie

```
myArr = [1, 5, 12]
```

[1, 5, 12]





User-defined functions



```
def printMonths():
   print("Jan, Feb, Mar")
def printDays():
   print("Mon, Tue, Wed")
                Mon, Tue, Wed
printDays()
printMonths()
                Jan, Feb, Mar
```



name





Parameters

```
Hello World
print("Hello World")
print("Test")
                       Test
                                                                   s = "This is a String"
                                                    s = "This is a String"
                                                    x = len(s)
                       16
print(len(s))
                                                    print(x)
```

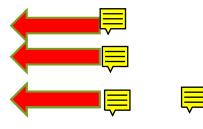


Parameters

```
def checkAge(age):
    if age < 18:
         return "Too Young"
    return "Accepted"
def main():
    name = input("Enter Name:" )
    age = int(input("Enter Age: "))
    print(name, "is", checkAge(age))
if __name__ == "__main__":
    # execute only if run as a script
    main()
```











22



Tom is Accepted







Local Variables

```
def checkAge(age):
     limit = 18
     if age < limit:</pre>
         return "Too Young"
     return "Accepted"
def main():
     name = input("Enter Name:" )
     age = int(input("Enter Age: "))
     print(name, "is", checkAge(age), limit
if __name__ == "__main__":
     # execute only if run as a script
     main()
```

```
def checkAge(age):
     limit = 18
     if age < limit:</pre>
          return "Too Young"
     return "Accepted"
def main():
    limit = "Finished"
     name = input("Enter Name:" )
     age = int(input("Enter Age: "))
     print(name, "is", checkAge(age), limit)
if __name__ == "__main__":
     # execute only if run as a scr:
     main()
```

Global Variables

```
def incrementAge(age):
    age += 1
    print(age)
                                 25
def main():
    age = 24
    incrementAge(age)
                                 24
    print(age)
if __name__ == "__main__":
    # execute only if run as a script
    main()
```

```
age = 24
def incrementAge():
    global age
    age += 1
    print(age)
                                  25
def main():
    incrementAge()
                                  25
    print(age)
if __name__ == "__main__":
    # execute only if run as a script
    main()
```