



AMC ENGINEERING COLLEGE
NAAC A+ ACCREDITED | NBA ACCREDITED
Banmerghatta road, Bengaluru, Karnataka 560083

Assignment 1

PYTHON PROGRAMMING

Submitted To:
PROF.RASHMI PARUTHI

Submitted By:
[REDACTED]
CSE 'D' SECTION
Vedashree M R

Veda01.ipynb

⏏ + ✂ 📄 📋 ▶ ■ ↺ ⏩ Code ▾

```
[7]: a=7
      if(a%2)==0:
          print("a is even")
      else :
          print("a is odd")
```

a is odd

```
[ ]:
```

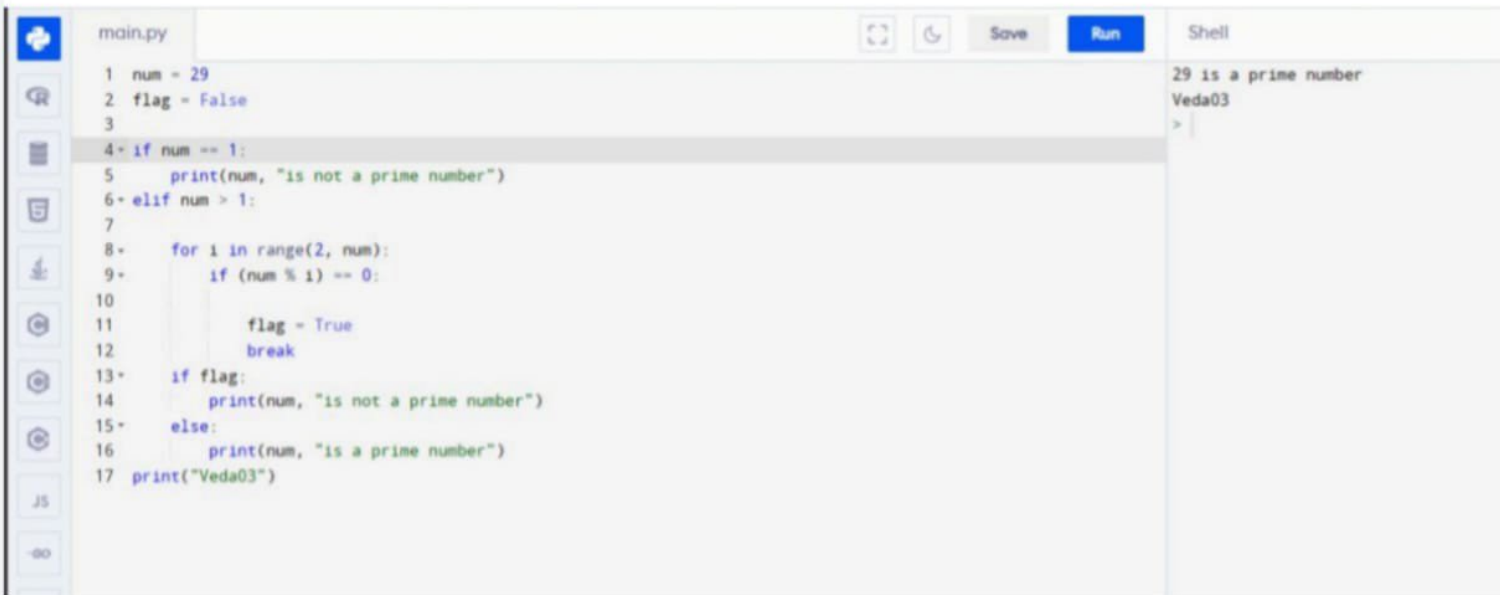
Veda02.ipynb

⏏ + ✂ 📄 📋 ▶ ■ ↺ ⏩ Code ▾

```
•[1]: n=5
      x=["positive"
          if n>0
          else "negative"
          if n<0
          else "zero" ]
      print(x)
```

['positive']


```
[ ]:
```



The image shows a Python IDE window titled 'main.py'. The code defines a function to check if a number is prime. It starts with `num = 29` and `flag = False`. A loop from 2 to `num` checks for divisibility. If a divisor is found, `flag` is set to `True` and the loop breaks. After the loop, it prints 'is not a prime number' if `flag` is `True`, and 'is a prime number' otherwise. The script also prints 'Veda03'.

```
1 num = 29
2 flag = False
3
4 if num == 1:
5     print(num, "is not a prime number")
6 elif num > 1:
7     for i in range(2, num):
8         if (num % i) == 0:
9             flag = True
10            break
11 if flag:
12     print(num, "is not a prime number")
13 else:
14     print(num, "is a prime number")
15 print("Veda03")
```

The Shell output on the right shows: `29 is a prime number` and `Veda03`.



The image shows a Jupyter Notebook window titled 'Vead04.ipynb'. The code defines a function `isPalindrome(s)` that returns `s == s[::-1]`. It then checks if the string 'malayalam' is a palindrome and prints 'Yes' or 'No'.

```
[1]: def isPalindrome(s):
      return s == s[::-1]

      s = "malayalam"

      ans = isPalindrome(s)

      if ans:
          print("Yes")
      else:
          print("No")
```

The output of the cell is 'Yes'.

Veda05.ipynb

📁 + ✂️ 📄 ▶️ ⏏️ ↺ ⏩ Code ▾

```
[2]: num1 = 12
      num2 = 8

      sum = num1 + num2

      print('The sum of {0} and {1} is {2}'.format(num1, num2, sum))
```

The sum of 12 and 8 is 20

[]:

Veda06.ipynb

📁 + ✂️ 📄 ▶️ ⏏️ ↺ ⏩ Code ▾

```
[3]: def add_num(a,b):
      sum = a + b
      return sum

      num1 = 5
      num2 = 7

      sum = add_num(num1, num2)

      print('The sum of numbers {0} and {1} is {2}'
            .format(num1, num2, sum))
```

The sum of numbers 5 and 7 is 12

[]:

Veda07.ipynb

+

Code

```
[7]: def maximum(a, b):  
      if a >= b:  
          return a  
      else:  
          return b  
      a = 10  
      b = 4  
      print(maximum(a,b))
```

10

[]:

Veda08.ipynb

+

Code

```
[6]: def minimum(a, b):  
      if a <= b:  
          return a  
      else:  
          return b  
      a = 10  
      b = 4  
      print(minimum(a,b))
```

4

[]:

Veda09.ipynb

⏏ + ✂ 📄 📋 ▶ ⏏ ↺ ⏏ ▶ Code ▾

```
[2]: n = 20
      num1 = 0
      num2 = 1
      next_number = num2
      count = 1

      while count <= n:
          print(next_number, end=" ")
          count += 1
          num1, num2 = num2, next_number
          next_number = num1 + num2
      print()
```

1 2 3 5 8 13 21 34 55 89 144 233 377 610 987 1597 2584 4181 6765 10946

[]:

Veda10.ipynb

⏏ + ✂ 📄 📋 ▶ ⏏ ↺ ⏏ ▶ Code ▾

```
[2]: n = 20
      fact = 1

      for i in range(1, n+1):
          fact = fact * i

      print("The factorial of 20 is : ", end="")
      print(fact)
```

The factorial of 20 is : 2432902008176640000

Veda11.ipynb

⏏ + ✂ 📄 📋 ▶ ⏏ ↺ ⏏ ▶▶ Code ▾

```
[2]: def gcd(a, b):  
      if b == 0:  
          return a  
      else:  
          return gcd(b, a % b)  
  
      num1 = 50  
      num2 = 68  
      print("The GCD of", num1, "and", num2, "is", gcd(num1, num2))
```

The GCD of 50 and 68 is 2

[]:

Veda12.ipynb

⏏ + ✂ 📄 📋 ▶ ⏏ ↺ ⏏ ▶▶ Code ▾

```
[2]: a = 20  
      b = 10  
  
      a, b = b, a  
  
      print("a =", a, "and b =", b)
```

a = 10 and b = 20

[]:

Veda13.ipynb

+

Code

```
[4]: num = "019876"
      reversed_num = num[::-1]
      print("The reversed number is:", reversed_num)
```

The reversed number is: 678910

[]:

main.py

```
1 import random
2 def guess_number():
3     print("Enter a random number:")
4     secret_number = random.randint(1,100)
5     while True:
6         guess = int(input("Guess the number (between 1 and 100): "))
7
8         if guess == secret_number:
9             print(" You guessed the correct number.")
10            break
11        elif guess < secret_number:
12            print("Too low! Try again.")
13        else:
14            print("Too high! Try again.")
15            print("veda14")
```



Save

Run

Shell

Guess the number (between 1 and 100): 67
Too high! Try again.
veda14
Guess the number (between 1 and 100): 98
Too high! Try again.
veda14
Guess the number (between 1 and 100): 50
Too high! Try again.
veda14
|