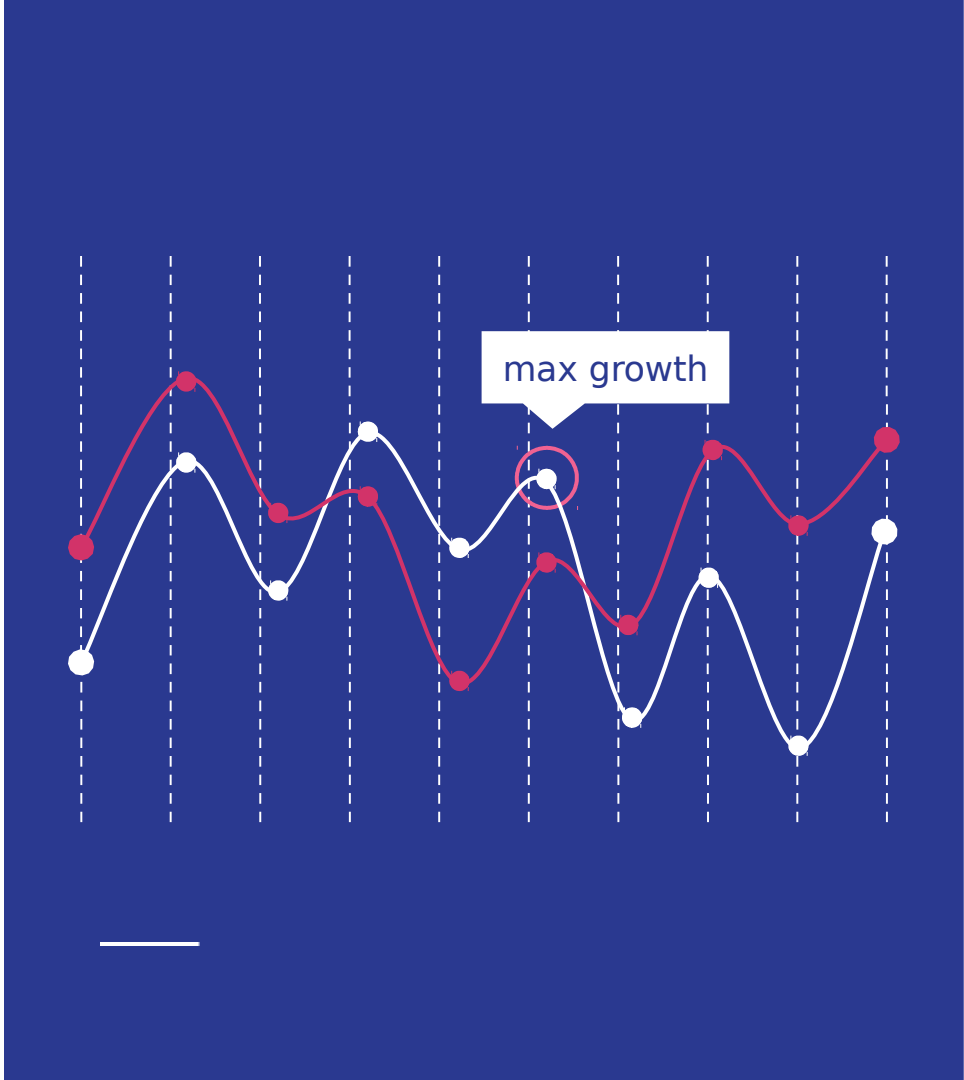
# Introduction to Computer Programming

## CMP 201 (2019/2020)

**SORINOLU, Babafemi Gabriel**



# **LESSON 1:**
# Getting Started

# Objectives

The aim of this lesson is to introduce the **python programming language**.

# Content(Week 1)

- **What is Python?** 
- Setting Up/Installation
- Uses of Python
- Getting Started! – Syntax, Comments
- Variables

# What is coding?

**Coding or programming is the art of writing instructions on our computer to perform a specific task.**

**Programs are specific sets of instructions executed on our computer .**

# What is python?

Python is a popular programming language. It was created by Guido van Rossum, and released in 1991.

## What can Python do?

- Python can be used on a server to create web applications.
- Python can be used alongside software to create workflows.
- Python can connect to database systems. It can also read and modify files.
- Python can be used to handle big data and perform complex mathematics.
- Python can be used for rapid prototyping, or for production-ready software development.

# Python

## (Getting Started)

# Installation:

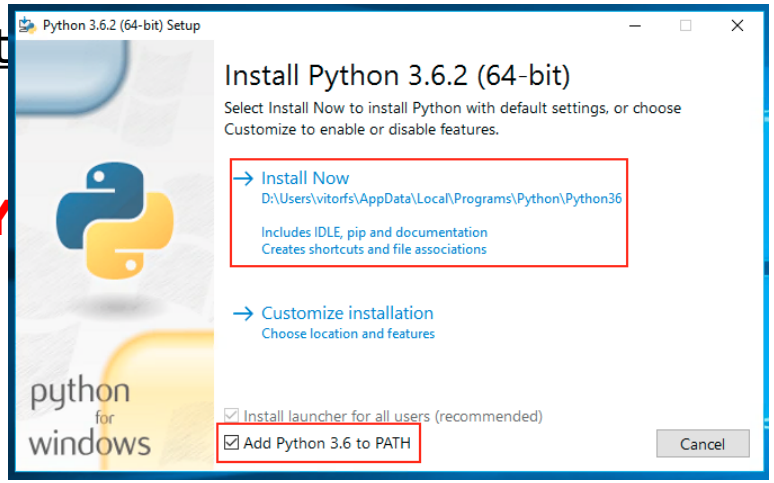
# Tools and Environment:

Python is an interpreted programming language, you can write Python (.py) files in a text editor and then put those files into the python interpreter to be executed.:

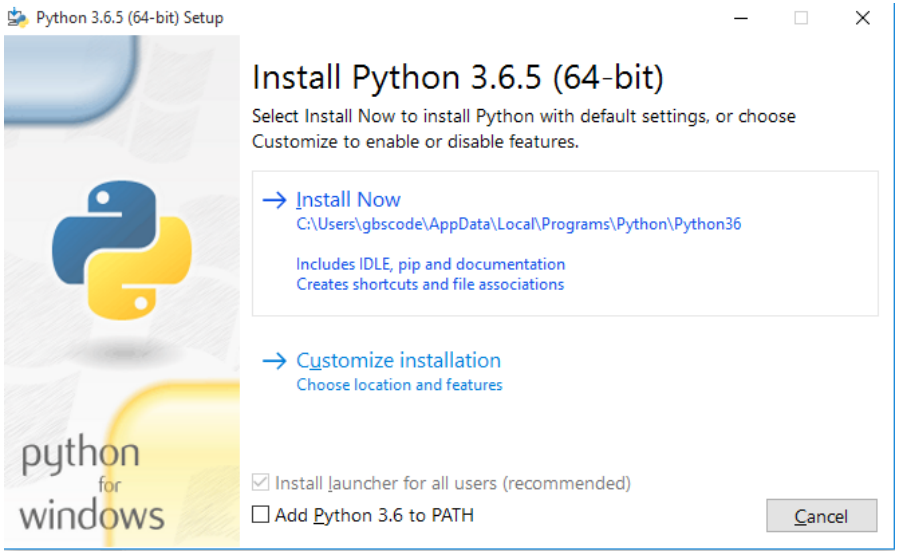
**Latest Python Version**: You can download version 3.7.4 from https://www.pyt

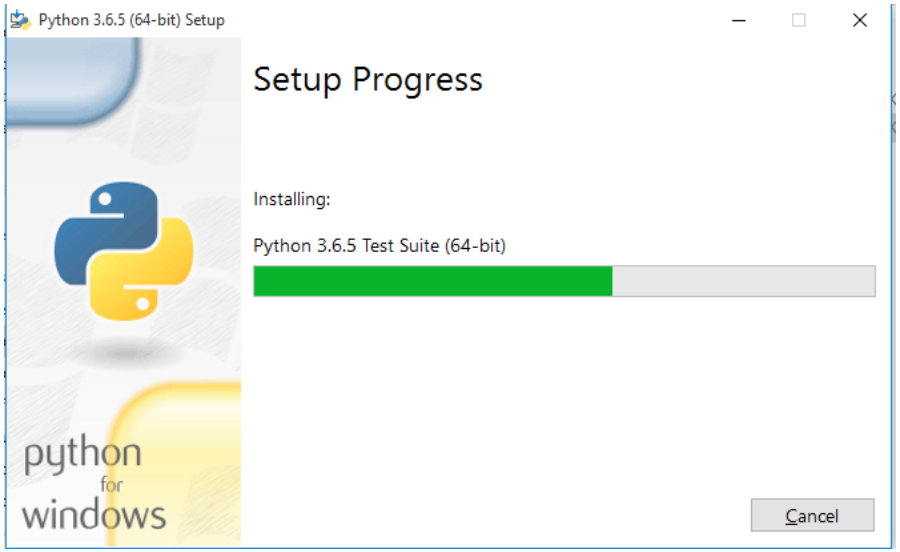
**PLEASE ENSURE PY**

**ADDED TO PATH**



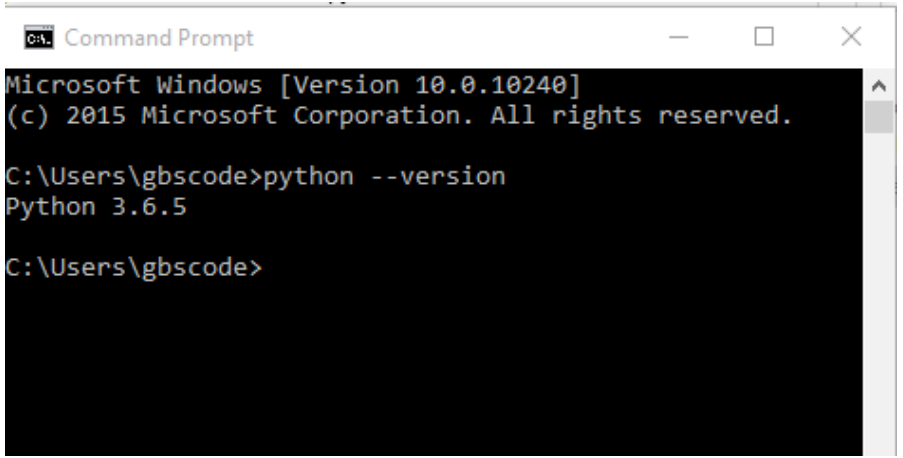
# Installing Python



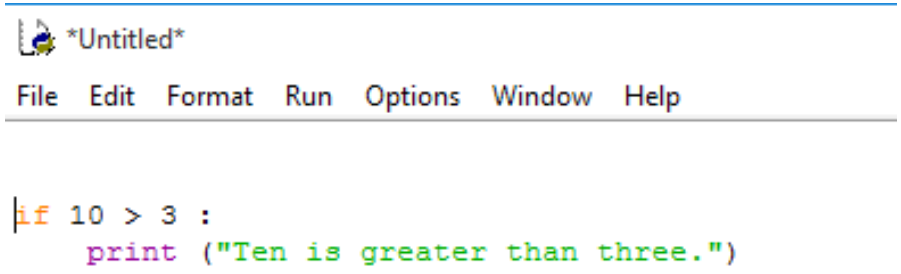
# Verifying Installation

To check if you have successfully installed python on
your Windows PC, search in the start bar for Python
or run the following on the Command Line (cmd.exe):



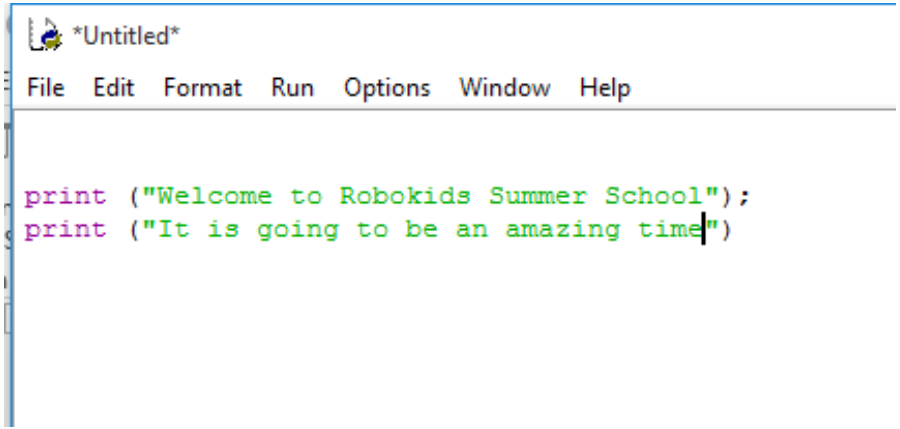
# Python Syntax:

Python uses indentation to indicate a block of code. Indentation is very important



# Python Syntax:

Using semi-colon at the end of a python statement is optional.



# Hello People!

## Writing our First Code

**Let us open our Python IDLE and get started!**

```
print ( " Hello World ")
print ( ' My name is John' )
```

# Remember:

- Case sensitivity
- Use of single or double quote
- Semi-colon is optional
- Output to screen using the **print** keyword

# Task

## To do in class

**PUZZLE:**

```
print ( " Hello World ")
print ( ' My name is John' )
```

**PUZZLE:**

```
print ( 55/11)
print ( ‘’55/11’ )
```

```
print ( 2 + 2 )
print ( 2 * 5  )
```

```
print ( 5 – 2 )
print (  12 / 4
        )
```

# Operators:

| | |
|---|---|
| 2+2 | Addition |
| 2-2 | Subtraction |
| 2*2 | Multiplication |
| 2/2 | Division |

The various math operators in python

```
print ( 10 / 3 )
print ( ‘ Hurray! I can code ’ )
```

# Accepting User Input:

- We can accept input from the user using the input() keyword.
- When accepting user input, we should store them in variables for future use.

Syntax

```
age= input('what is your age?')
```

# Comments:

- Comments can be used to explain python code
- Comments can be used to make the code more readable
- Comments can be used to prevent execution when testing code

# Creating a Comment:

Comments start with a #, and Python will ignore them.

```
#This is a comment

#The code below displays hello world

print ( ‘ Hello world’ )
```

# Creating a Comment(2):

Comments can be placed at the end of a line, and Python will ignore the rest of the line.

```
print ( ‘ Hello world’ ) #This is a comment
```

# Creating a Comment(3):

Comments can be used to prevent Python from executing a code.

```
#print ( ‘ Hello world’ )

print ( ‘ Wow! This is fun.’ )
```

# Multi line Comments:

To add a multi line Comment, you could insert a # for each line.

```
#This is a comment on line 1

#and another on line 2

#and one more on line 3
```

# Multi line Comments(2):

You could also use the multiline string (triple quotes) and place your comments inside it.

```
"""
Comments can be placed here
"""
Line 1 :
```

# PUZZLE:

```
# This is a comment

#Now back to the puzzle

text = "# Is this a comment ?"

print ( text )
```

# What are Variables?

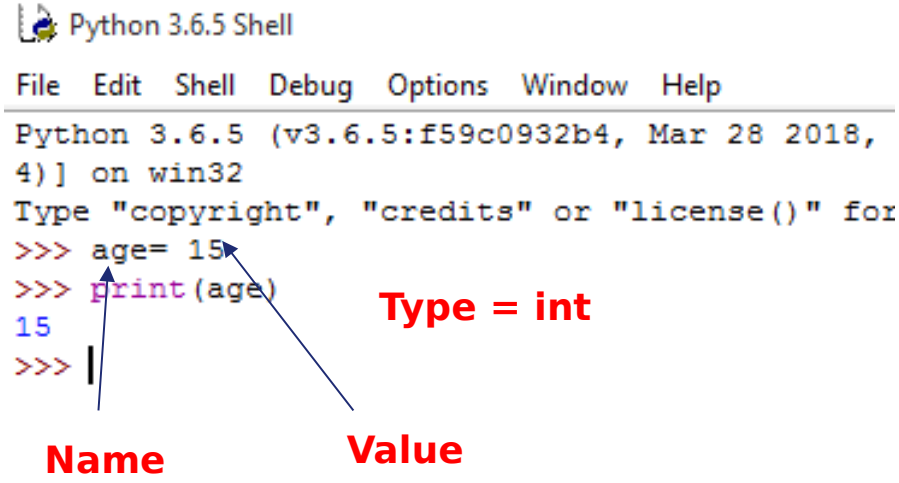
# Variables:

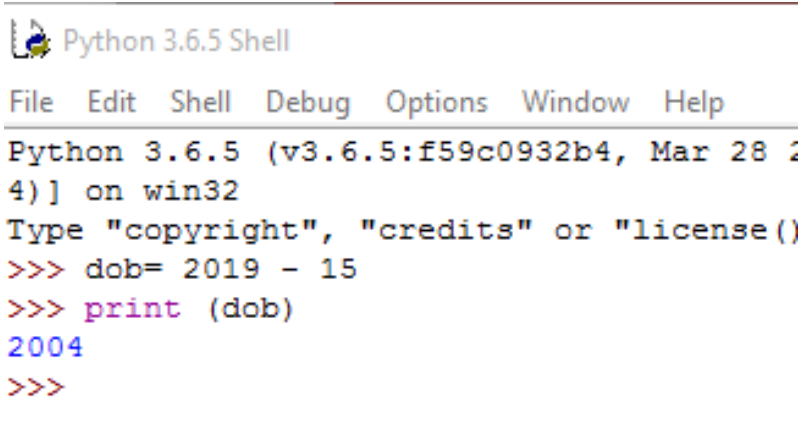
- Think of variables like a box that can hold values
- Variables are containers used to store a value.
- Variable comprises of 3 components – name, value and data type.

# Variables:

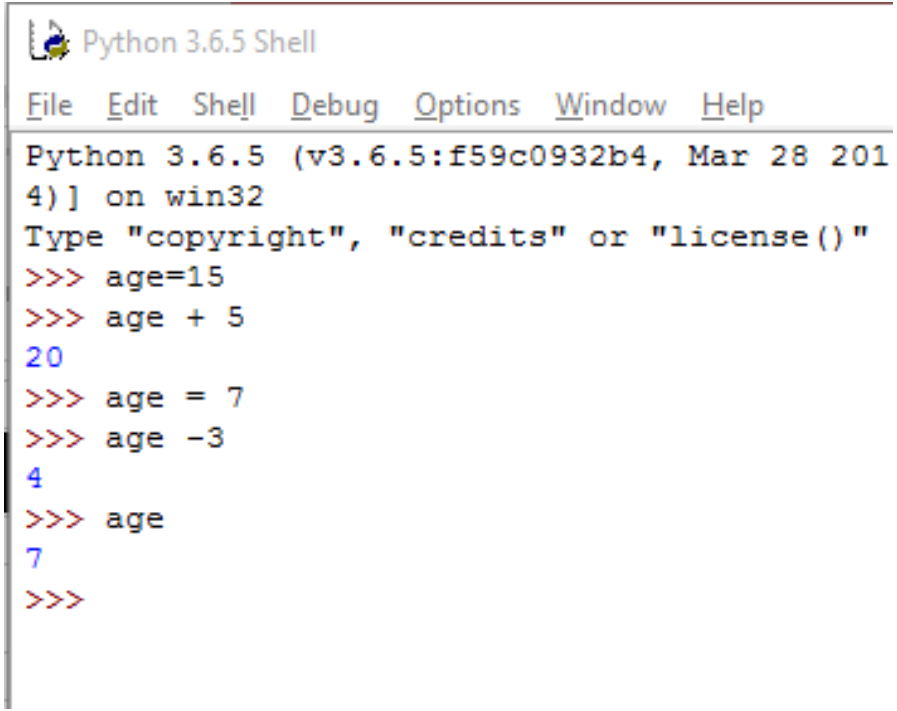
- You can store values inside variables with the = sign (called the assignment operator).
- For example, to store the value 15 in a variable named “age", enter age= 15 into the shell:

# Variables:


Python 3.6.5 Shell
File Edit Shell Debug Options Window Help
Python 3.6.5 (v3.6.5:f59c0932b4, Mar 28 2018,
4)] on win32
Type "copyright", "credits" or "license()" for
>>> age= 15
>>> print(age)
15
>>>
Type = int
Name
Value


Python 3.6.5 Shell
File Edit Shell Debug Options Window Help
Python 3.6.5 (v3.6.5:f59c0932b4, Mar 28 2
4)] on win32
Type "copyright", "credits" or "license()
>>> dob= 2019 - 15
>>> print (dob)
2004
>>>

# Variables:


Python 3.6.5 Shell
File Edit Shell Debug Options Window Help
Python 3.6.5 (v3.6.5:f59c0932b4, Mar 28 201
4)] on win32
Type "copyright", "credits" or "license()"
>>> age=15
>>> age + 5
20
>>> age = 7
>>> age -3
4
>>> age
7
>>>

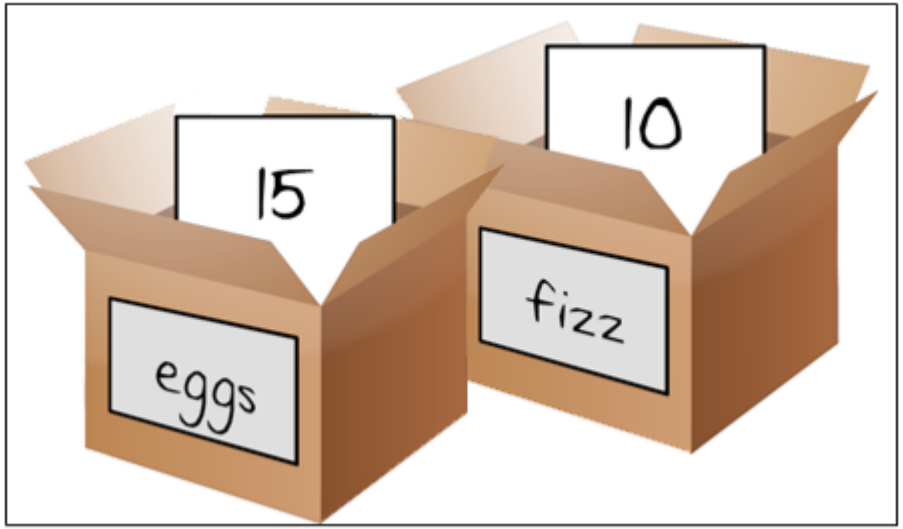
# Remember:

- Case sensitivity
- Make sure you declare before using a variable
- We assign a value to a variable using the = operator

# Using more than one Variables:

When we program we won't always want to be limited to only one variable. Often we'll need to use multiple variables

# Using more than one Variables:




test.py - C:/Users/gbscode/Desktop/CSC 7

File Edit Format Run Options Windo

```
fizz = 10
eggs = 15

print ('Fizz : ',fizz)
print ('Eggs : ', eggs)
```

**Now the fizz variable has 10 inside it, and eggs has 15 inside it.**

# PUZZLE:

```
a= 50

b= 10

c= a - b

print (c)

print (a)

print (b)
```

# PUZZLE:

```
x= 50 * 2 + (60 – 20 ) /4

print (x)
```

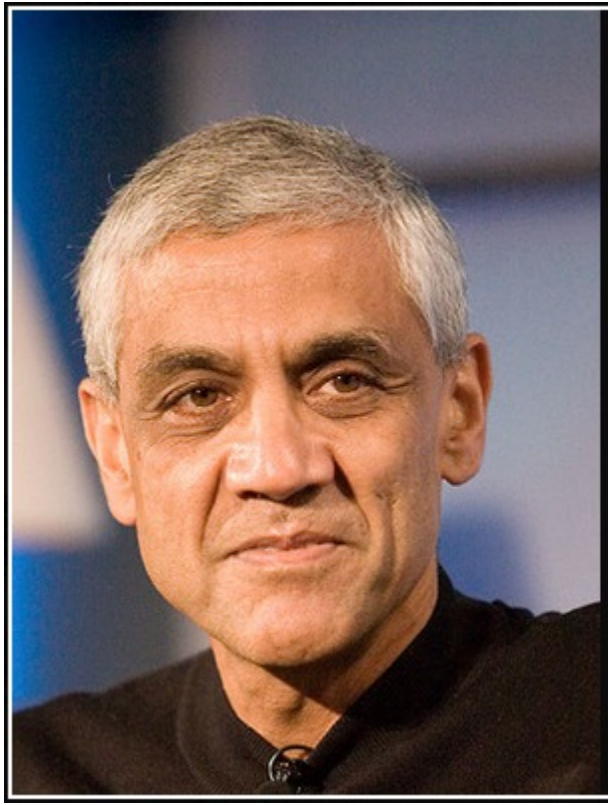
# PUZZLE:

print (school)

# EXERCISE

(Uniqueness in code earns extra credit).

1. **Each question should be kept in a single .py file**
2. **then all zipped in a file**
3. **with your matric No. as name of the file**



Doctors can be replaced by software – 80% of them can. I'd much rather have a good machine learning system diagnose my disease than the median or average doctor.

— Vinod Khosla —

AZ QUOTES