Infosys

Big Data

→ Terabyte's of data ⇒ How to store?

Data size ⇒ ↑

Big Expansion → Lot of data.

* Framework to organize data

BIG DATA
- → Scalable
- → Store any amt of data
- → Allow me write any no. of queries
- → Processing any no. of data.

I] VOLUME →

II] VARIETY → Images, videos etc.

III] VELOCITY → Speed (what rate data is growing)

IV] VERACITY → (uncertainity of data)
- → Consider only relevant data

* Session Plan

* Structured data

→ Organised data ⇒ RDBMS, Excel

* Semi Structured → HTML, XML

* Unstructured ⇒ Media files
→ Data from sensors & physical device.
→ Videos, Images.

* BIG DATA

Tera, Peta, 1Exa, Zeta, Yotta

* Data Analysis
Business Insights

* Sentiment Analysis ⇒ ML Algo.
→ Analyse the Sentiment of people
→ Predict the outcome.

Challenge

→ Storage, processing ⇒ Data.

* Technologies

↳ Schedule & run parallel tasks. → (Reduces time

↳ Coordinate Computing tasks.

↳ Aggregate results.

↳ Partition data, back up data.

(1) Parallel processing | RDBMS → SQOOP

(2) SQL

HADOOP

* HADOOP Framework

HDFS Storage

Java.

MapReduce

Process data.

Framework to process data across multiple machines

→ Map

→ Reduce

* MetaData → Data above data

* Master Data → Manage all slave machines.

128 MB ⇒ 1 Blocks → Slave.
2 Blocks.

↳ Back up Data ⇒ Replication factor.
Default = 3
↳ Back up data.

Distributed Environment Architecture.

Hadoop Distributed Files System ⇒ HDFS

<the, 1>
<is, 1> ⇒ <the, [1, 1....]
→ Shuffling.

SPARK → Faster than MapReduce.

Hive → SQL

Courses ⇒ Intro to Big Data.

Cassandra Developer

* PYTHON — 13th Aug, 2018

⇒ Version Compatibility. Version → New features

→ 2008 → Python 2.6 & Py 3.0, 3.1

2010 → Py 2.7 (LTS)

Py 2.x Version is different from 3.x

* Agile → Quick time to market
→ Respond to Changes.

Customer Collaboration → Cost of rework Changes in SDLC
Ind & Interaction / is very high.
er Contract negot. → Req. needs to be fixed in the beginning

→ Py 3.x is not backward Compatible.
→ Does not support Syntax of 2.x

2015 ⇒ Py 35 3.7.0

Python.org | Pythontutor.com → Learn.

Py → Cross platform
→ No Datatype, Everything is object
→ supports object oriented concepts
GUI, DB, Image processing, ML, AI,
→ Data science & Analysis.

Python

→ No datatype → Dynamically allocate type to variables based on value assigned to the variable.

(b) No., String, Boolean & object:

→ Case Sensitive.

→ No need for ;

```
a = 10
a
print(a)
(10) →
```

I/P

O/P

print("The value of a is", a)

Py → Extension → .Py / .Pyw

Scrum → Dev projects

Kanban → Maintenance projects.
(DevOps)