Apache Spark Project
=======================

This is the part where most of the candidates get stuck.

Project

- easy (transformations, performance tuning)
- hard (if you are not prepared)

most of the candidates get stuck with these things...

1. to explain about a realisting project idea based on your domain
banking, retail, health care, insurance, finanance, sales

2. how does agile methodology works

3. how do you deploy your code - CICD part
dev, stage, prod

4. as part of cleaning what all you did
   - handling duplicates
   - dealing with nulls
   - handling missing values
   - changing the datatypes

5. how did you do unit testing
   - pytest

6. what transformations did you use as part of your project

7. SCD (slowly changing dimensions)
customer address - Bangalore time1
Hyderabad time2

8. how did you do the estimation of resources required for your project

9. what infrastructure you used ? on cloud / on premise

10. how much data you deal on a day to day basis

11. what is your role in your big data project

12. what was your cluster size

the hard part is a good problem statement + Process that is followed

A few Problem statements from different domains
===================

1. Reporting -

cisco there were sales executives who used to go to customers...

cisco is trying to sell a telepresence system to walmart

cisco routers.. not been installed..

P1 ticket which is pending for long..

customer 360 -

ucrm cases - functional tickets

tac cases - technical tickets

other datasets....

360 degree view of the customers

multiple data sources

Raw files / Databases -> ingestion -> Datalake -> Processed using spark -> Database/Datawarehouse/NoSQL

we could have ran these queries on database itself, then why we used all of the big data technologies...

your database is handling day to day transactions, you do not want to overburden your database with crunching of data.

2. MDM (master data management)

single source of truth..

- customer
- employee
- vendor
- product
- location
- reference

customer sumit purchased a product macbook pro at apple store phoenix marketcity bangalore at 8 pm on 13th june for $1000 (80000 Rs)

customer - sumit
product - macbook pro
location - bangalore
vendor - apple store
reference data

we are developing a MDM system and adding few entities to it.

multiple sources -> clean the data -> transform

centralized repository of data which is most up to date and all teams can use it.

MDM (master data management) - your company has a MDM

we have to use this data to create certain reports to analyze customer churn

3. Finance Domain -

Lending Club

borrowers

investors

risk factor - my current salary, last 3 months salary slips, I own a house or not, whether I have defaulted on my previous loans...

If the risk factor is high - the interest % will be really high

6% to 20%

4. co-brand cards

banking related...

Airtel axis bank credit card
sbi vistara card

analysis done (Product analytics)

5. Healthcare - Improve patient outcomes and reduce hospital readmissions through predictive analytics

Electronic health records - history, treatment, outcomes
wearable devices - fitbit, apple watch
genomics data - personalized treatment based on genetics
social media - patient feedback, experience and sentiments

Data Cleaning - missing values, outliers and errors
Data transformation

6. Anaplan for sales:

GTM team (GO to market strategy)

A go to market (GTM) stragegy is a plan that details how an organization can engage with customers to convice them to buy their product or service

=> Anaplan for sales is designed to empower your organization go to market strategy and help sales leaders make better decisions..

collected data from many sources

ADF -> Pyspark (Cleaning -> transforming) -> Anaplan

Cleaning , transformation, aggregations , optimizations - we know most of it.

Agile methodology works

what is SCD and how to implement

Deployement - CICD

Unit testing

Implement logging

Modular architecture

Configuration file

Parameterize the pipeline (templatize)

How to estimate the cluster resources

Agile Methodology
==================

if you are a developer, how do you get tasks to work on?

Agile & Waterfall
====================

big project - 6 months

Agile - Sprints (1 week / 2 week)

6 sprints (2 weeks)

12 weeks (3 months)

we develop the project incrementally

2 weeks something which can be demo is developed...

stories

can you implement logging for this project - 2 points

can you bring customers data to the data lake - 3 points as a developer you would be assigned a few stories

each story has story points

- 1 (in a few hours)

- 2 (1 day)

- 3 (2 days)

- 5 (3-4 days)

Initiative -> Epics -> Stories -> tasks -> sub tasks

Data Storage - Create a Data lake to store raw, cleaned and processed data for future analysis.

Data Ingestion - create a scheduled job to extract data from an external database and load it to our data lake.

Data Transformation - create a data transformation pipeline to clean and standardize incoming customer data.
Enrich the data with additional attributes.

Data Processing - Design and implement a batch pipeline to aggregate monthly sales data.

real time data - create a real time streaming pipeline for monitoring website traffic.

Data Quality & Validation - Implement automated data validation checks to ensure the integrity of incoming data.

Data Documentaion - Document the data Schemas for the customer database to facilitate data analysis.
Maintain up to date documentation for the ETL processes.

Testing - write unit tests for data transformation functions to ensure that logic is correct.

Performance optimization - fine tune the jobs to improve resource utilization and minimize processing time.

Deployment & scaling - setup automated deployment pipleines for data processing jobs to streamline deployment to production.

Design an auto scaling infra to handle increased data volumes during peak hours...

============

Agile
======

Sprints (2 weeks)

SCRUM (a project management technique to deliver a project in agile manner)

There are 4 different types of people in the project

1. Product owner (Project Manager)
2. Scrum master (plays a role of driving the things)
3. Developers
4. Stake holders

scrum team (1,2,3)

whenever a project comes...

Lending Club - Risk Analysis

epic 1 - Develop a quick Prototype

epic 2 - Make it production ready

epic 3 - improve the performance

Product Owner who is responsible to create Backlogs

The product owner talks to various people and comes us with stories and adds them as part of Backlogs..

1. Sprint Planning - The product owner brings the product backlog to discuss with the dev team. The scrum team does effort on story point estimation. The product backlog must contain all the details necessary for this estimation..

The Dev team, Scrum Master, Product Owner

At the beginning of each Sprint

2 hours meeting.

2. Daily Stand-up - 10 mins call , a very quick call...

blockers, what have you worked on yesterday and what are you planning to work on today. Generally happens in the morning.

Dev team, Scrum Master, Product Owner (optional)

3. Sprint Review - Showcase the teams completed work and gather feedback from stakeholders.

Scrum Master drives this.

Dev team, Product owner, Scrum Master, Stakeholders (management, customers)

1 hour meeting

4. Sprint Retrospective - you look back on work that has been done in this sprint. what went well and what could have been done better.

Dev team, scrum master and product owner

1 hour

Banking/Finance - Pyspark

Money Lending Institutions
==============================

Traditional lending institutes - Bank

Alternate lending institutes

Zest money

Lendbox - peer to peer

prosper

propelld - education

Fibe

Liquiloans

Grayquest

Eduvanz - education

lending club - peer to peer

they give loan from their side...

- institute give the loan / borrower - institue
- peer to peer / borrower - investor/lender

Pros -

- less paper work
- long term loans
- even with low credit score we can get loan
- quick process
- hassle free

cons

- high interest rates
- can be risky for the investors

investors get a interest percentage (6% to 36% anually)

36 months - 60 months (repayment)

consider one of these financial institute is your client, and they send you the data so that our data engineering team can clean it, process it and give it back to be consumed by data analytics team.

you work for a consumer finance company which specializes in lending various types of loans to urban customers. the company has to either approve or disapprove a loan based on application.

data engineering team should clean the data so that the other teams can use the cleaned data and also the data engineering team should calculate the risk score based on which the company can decide whether to approve or disapprove.

=> if the applicant is likely to pay the loan, then not approving the loan results in business loss.

=> if the applicant is not likely to repay the loan, then approving the loan may lead to a financial loss to the company

clean

process - credit score (certain rules)

lending club dataset - 1.7 GB

1 file (csv format)

118 columns

2+ million records

customers_data (borrowers details)
================
member_id, emp_title, emp_length, home_ownership, annual_inc, addr_state, zip_code, country, grade, sub_grade,verification_status, tot_hi_cred_lim,application_type,annual_inc_joint,verification_status_joint

loans_data
==========
loan_id,member_id,loan_amnt,funded_amnt,term,int_rate,installment,issue_d,loan_status,purpose,title

loan_repayments
================
loan_id,total_rec_prncp,total_rec_int,total_rec_late_fee,total_pymnt,last_pymnt_amnt,last_pymnt_d,next_pymnt_d

loan_defaulters
================
member_id,delinq_2yrs,delinq_amnt,pub_rec,pub_rec_bankruptcies,inq_last_6mths,total_rec_late_fee, mths_since_last_delinq, mths_since_last_record

emp_title - engineer

emp_length - 3

home_ownership - own

annual_inc - 50000

zip_code - 670xx

addr_state - CA

grade - B

sub_grade - B

verfication_status - approved

We can use a hash function which can take the values of the above 9 columns and generate a hash code sha2 -

input
======

emp_title - engineer

emp_length - 3

home_ownership - own

annual_inc - 50000

zip_code - 670xx

addr_state - CA

grade - B

sub_grade - B

verfication_status - approved

output
=======

a unique value (32 characters)

if you want to generate a new column in a pyspark dataframe then you need to use

withColumn

```
new_df =
raw_df.withColumn("name_sha2",sha2(concat_ws("||",*["emp_title","emp_length","home_ownership","annual_inc","zip_code","addr_state","grade","sub_grade","verification_status"]),256))
```

total records is 2260701

unique members are 2257384

===================

customers_data (borrowers details)
=================
member_id, emp_title, emp_length, home_ownership, annual_inc, addr_state, zip_code, country, grade, sub_grade,verification_status, tot_hi_cred_lim,application_type,annual_inc_joint,verification_status_joint

loans_data
===========
loan_id,member_id,loan_amnt,funded_amnt,term,int_rate,installment,issue_d, loan_status,purpose,title

loan_repayments
=================
loan_id,total_rec_prncp,total_rec_int,total_rec_late_fee,total_pymnt,last_pymnt_amnt,last_pymnt_d,next_pymnt_d

loan_defaulters
=================
member_id,delinq_2yrs,delinq_amnt,pub_rec,pub_rec_bankruptcies,inq_last_6mths,total_rec_late_fee, mths_since_last_delinq, mths_since_last_record

Cleaning of data is really important

Cleaning & Processing

cleaning of customers data
=============================

1. create a dataframe with proper datatypes

2. Rename a few columns
annual_inc -> annual_income
addr_state -> address_state
zip_cod -> address_zipcode
country -> address_country
tot_hi_cred_lim -> total_high_credit_limit
annual_inc_joint -> join_annual_income

3. insert a new column named as ingestion date (current time)

4. remove complete duplicate rows

5. remove the rows where annual_income is null

6. convert emp_length to integer

7. we need to replace all the nulls in emp_length column with average of this column

8. clean the address_state (it should be 2 characters only), replace all others with NA

write the cleaned customers data in cleaned folder in hdfs.

```
withColumnRenamed("annual_inc","annual_income")
withColumnRenamed("addr_state","address_state")
withColumnRenamed("zip_code","address_zipcode")
withColumnRenamed("country","address_country")
withColumnRenamed("tot_hi_cred_lim","total_high_credit_limit")
withColumnRenamed("annual_inc_joint","join_annual_income")
```

```
from pyspark.sql.functions import current_timestamp
withColumn("ingest_date",current_timestamp())
```

A B C
A B C

Cleaning of Loans Data
=======================

raw -> do cleaning -> cleaned

```
loans_schema = 'loan_id string,member_id string,loan_amount
float,funded_amount float,loan_term_months string,interest_rate
float,monthly_installment float,issue_date string,loan_status
string,loan_purpose string,loan_title string'
```

```
columns_to_check = ["loan_amount", "funded_amount", "loan_term_months",
"interest_rate",
"monthly_installment","issue_date","loan_status","loan_purpose"]
```

```
loan_purpose_lookup = ["debt_consolidation", "credit_card",
"home_improvement", "other", "major_purchase", "medical", "small_business",
"car", "vacation", "moving", "house", "wedding", "renewable_energy",
"educational"]
```

Cleaning of loans_repayment data
===================================

customers (borrowers)
loans data

```
'loan_id string,total_principal_received float,total_interest_received
float,total_late_fee_received float,total_payment_received
float,last_payment_amount float,last_payment_date string,next_payment_date
string'
```

```
columns_to_check =
["total_principal_received","total_interest_received","total_late_fee_received","
total_payment_received","last_payment_amount"]
```

```
loans_payments_ldate_fixed_df = loans_payments_fixed2_df.withColumn(
    "last_payment_date",
    when(
        (col("last_payment_date") == 0.0),
        None
        ).otherwise(col("last_payment_date"))
)
```