

Guided Project

PySpark Foundations: Process, analyze, and summarize data

Estimated Time
45 minutes

Meet Olayinka Arimoro

- 5 years experience analyzing data.
- 3+ years experience building machine learning models working with health data.
- Top-rated Coursera guided project instructor with 40 courses and about 100K learners.



Recommended Background

Basic Python Programming

- Knowledge of Python syntax
- Basic data structures, and data frame operations like filtering, grouping, and summarizing data.



Project Outcome

By the end of this project-based course, learners will be able to process, analyze, and summarize large data using PySpark, including performing data cleaning and aggregation for data insights.

Learning Objectives

- Process large datasets using PySpark, including data loading, cleaning, and preprocessing.
- Perform data exploration and visualization using dataframe operations.
- Perform data aggregation and summarization using PySpark and dataframe operations.

Apache Spark

Key features



Batch/streaming data

Unify the processing of your data in batches and real-time streaming, using your preferred language: Python, SQL, Scala, Java or R.



SQL analytics

Execute fast, distributed ANSI SQL queries for dashboarding and ad-hoc reporting. Runs faster than most data warehouses.



Data science at scale

Perform Exploratory Data Analysis (EDA) on petabyte-scale data without having to resort to downsampling



Machine learning

Train machine learning algorithms on a laptop and use the same code to scale to fault-tolerant clusters of thousands of machines.

PySpark

- PySpark provides an interface for Apache Spark in Python.
- You can write Python and SQL-like commands to manipulate and analyze data in a distributed processing environment.

Project Scenario


Your Role: Entry-level data analyst/scientist

In this project, you will take on the role of a junior or entry-level data analyst/scientist and will use the employees/salaries data to perform analysis that covers key areas such as employee distribution across departments, average salaries, and age demographics. These analyses will provide key decision-makers with insight on how to compensate, retain and hire.

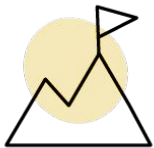
Task 1

Set up and overview of the project

You will get an overview of the project and set up the building blocks for the project.

Two decorative yellow arcs are positioned on the right side of the slide, curving upwards and outwards.

Task Summary



Set up and overview of the project



Key Takeaways

- PySpark is an interface for Apache Spark in Python to manipulate and analyze data in a distributed processing environment.
- **SparkSession.builder** creates an entry point to using PySpark; **appName()** names the Spark application; **getOrCreate()** retrieves an existing Spark session or creates a new one.

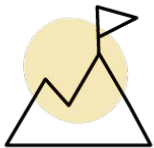
Task 2

Load the data

You will load the employees.csv and updated_salaries.csv data.

Two decorative yellow curved lines are positioned in the bottom right corner of the slide, adding a modern, abstract touch to the design.

Task Summary



Load the data




Key Takeaways

- **spark.read.csv:** a method in Spark that reads a CSV file into a data frame.
- **header=True:** specifies that the first row of the CSV file contains the header (column names)
- **inferSchema=True:** tells Spark to automatically infer the data types of each column based on the values in the CSV file.

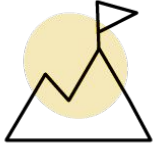
Task 3

Clean and process data

You will perform quick data cleaning by converting columns to proper data types.

Two decorative yellow curved lines are positioned in the bottom right corner of the slide. The upper line is a large, shallow arc that starts near the bottom center and curves towards the right edge. Below it is a smaller, more pronounced arc that also starts near the bottom center and curves towards the right edge.

Task Summary



Clean and process data




Key Takeaways

- Formatting allows for standardization and normalization of data. It aids in error detection and data cleaning, setting the stage for reliable data analytics.

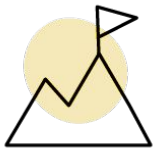
Task 4

Explore the data

You will explore the salaries data by computing summary statistics and visualizing the salary column.

Two decorative yellow arcs are positioned in the bottom right corner of the slide. The larger arc is in the background, and a smaller arc is in the foreground, both curving upwards and to the right.

Task Summary



Explore the data



Key Takeaways

- Data exploration, one of the first steps in data preparation, is a way to get to know data before working with it.
- **toPandas()** converts the Spark data frame to a Pandas data frame.

Practice Activity

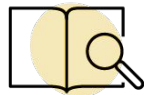


**This task is optional and ungraded.
The goal is to check your understanding.**

Practice Activity

In this practice task, you will explore the employees data. To complete this practice task, please follow the instructions below:

- Create a sum of missing values per column in the employees data.
- Count the number of rows in the employees data.
- Count the number of unique first names in the employees data.



Things to Note

- When returning the counts, you may decide to use or not use the f-string format to print the return.



Pro Tip


- ★ For this activity, you may review the task on “**Data exploration**”

(Note to the learner: Pause the video to complete the task and unpause to see the solution once the task is complete.)

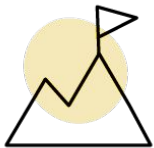
Task 5

Aggregate and summarize the data

You will perform data aggregation and summarization using the salaries data.

Two decorative yellow curved lines are positioned on the right side of the slide, starting from the bottom and curving upwards and outwards.

Task Summary



Aggregate and summarize the data




Key Takeaways

- The **groupBy** operation allows you to perform aggregate functions (such as sum, count, max, min, etc.) on grouped data, based on one or more columns.
- The **agg()** function is used to apply aggregate functions to the grouped data. You would typically pass functions inside **agg()** to specify what operation to perform on the grouped data.

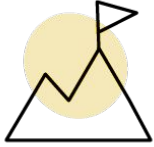
Task 6

Join the data sets

You will join the salaries and employees data using the employees number.

Two decorative yellow curved lines are positioned in the bottom right corner of the slide. The upper line is a large, shallow arc that starts near the bottom center and curves towards the right edge. Below it is a smaller, more pronounced arc that also starts near the bottom center and curves towards the right edge.

Task Summary



Join the data sets



Key Takeaways

- Use left join when you need all records from the left table and the matching records from the right table. Unmatched records from the right table will have NULL values.

**Congratulations on
completing your Guided
Project!**

Next steps?



Thank you!!!



Cumulative Activity



This activity is optional and ungraded.
The goal is for you to apply the knowledge and skills
learned within this Guided Project to boost your
confidence.

Cumulative Activity Scenario

As a junior data analyst at a growing company, you are tasked with analyzing employee retention. Your aim is to find departments with the highest amount of employees that have worked longer than ten years. This will assist HR in enhancing employees' engagement and retention strategies. To complete this activity, you will use the employee dataset and create a data frame with the employee counts in each department for a period over ten years (calculated by `from_date` and `to_date`). Finally, you'll visualize how long-term employees are spread across departments via a bar chart.



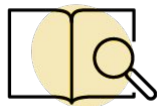
Your Task

1. Calculate the years worked based on the difference between `'to_date'` and `'from_date'`.
2. Group by `emp_no` and `dept_no` to sum the years worked.
3. Filter employees who have worked more than 10 years.
4. Group by department and count distinct employees who worked more than 5 years.
5. Create a bar chart to visualize the distribution of long-term employees across departments.

Cumulative Activity

Given that we have repeated rows for employees, where each row represents a period of employment with corresponding dates (from_date and to_date). To solve this, you can:

- Aggregate the years worked by each employee over multiple periods.
- That is, after you have calculated the number of years worked for each row (difference between from_date and to_date), group by emp_no and dept_no, and sum the years worked across all periods.



Things to Note

- Make sure that you convert the Spark data frame to Pandas before creating the bar chart.



Pro Tip

- ★ Review the task titled “**aggregate and summarize data**”

(Note to the learner: Pause the video to complete the task and unpause to see the solution once the task is complete.)

Thank you!!!

