TOP 30 Python Important Functions:

for interview preparation as well as for company use.

1. `SUM()`

Explanation: Calculates the sum of a column or expression.

Example:

```
TotalSales = SUM(Sales[Amount])
```

Rating: Interview Prep: 10/10 | Company Use: 10/10

---

2. `AVERAGE()`

Explanation: Calculates the average (arithmetic mean) of a column or expression.

Example:

```
AverageSales = AVERAGE(Sales[Amount])
```

Rating: Interview Prep: 9/10 | Company Use: 10/10

---

3. `COUNT()`

Explanation: Counts the number of rows in a column.

Example:

```
OrderCount = COUNT(Orders[OrderID])
```

Rating: Interview Prep: 9/10 | Company Use: 10/10

---

4. `COUNTA()`

Explanation: Counts the number of non-blank rows in a column.

Example:

```
NonEmptyProductCount = COUNTA(Products[ProductName])
```

Rating: Interview Prep: 9/10 | Company Use: 10/10

---

5. `COUNTROWS()`

Explanation: Counts the number of rows in a table.

Example:

```
TotalOrders = COUNTROWS(Orders)
```

Rating: Interview Prep: 10/10 | Company Use: 10/10

---

6. `DISTINCTCOUNT()`

Explanation: Counts the number of distinct values in a column.

Example:

```
UniqueCustomerCount = DISTINCTCOUNT(Sales[CustomerID])
```

Rating: Interview Prep: 10/10 | Company Use: 10/10

---

7. `SUMX()`

Explanation: Iterates over a table and evaluates an expression for each row, then sums the results.

Example:

```
TotalRevenue = SUMX(Sales, Sales[Quantity] * Sales[Price])
```

Rating: Interview Prep: 10/10 | Company Use: 10/10

---

8. `AVERAGEX()`

Explanation: Iterates over a table and calculates the average of an expression for each row.

Example:

```
AverageRevenuePerOrder = AVERAGEX(Sales, Sales[Quantity] * Sales[Price])
```

Rating: Interview Prep: 9/10 | Company Use: 10/10

---

9. `IF()`

Explanation: Checks a condition and returns one value if true, and another value if false.

Example:

```
HighRevenue = IF(Sales[Amount] > 1000, "High", "Low")
```

Rating: Interview Prep: 10/10 | Company Use: 10/10

---

10. `SWITCH()`

Explanation: Evaluates an expression against a list of values and returns the corresponding result.

Example:

```
CategoryLabel = SWITCH(
    TRUE(),
    Sales[Amount] > 1000, "High",
    Sales[Amount] > 500, "Medium",
    "Low"
)
```

Rating: Interview Prep: 9/10 | Company Use: 10/10

---

11. `RELATED()`

Explanation: Retrieves a related value from another table using a relationship.

Example:

```
ProductCategory = RELATED(Products[Category])
```

Rating: Interview Prep: 9/10 | Company Use: 10/10

---

12. `LOOKUPVALUE()`

Explanation: Retrieves the value from a table for the row that matches a given set of criteria.

Example:

```
ProductPrice = LOOKUPVALUE(Products[Price], Products[ProductID],
Sales[ProductID])
```

Rating: Interview Prep: 10/10 | Company Use: 10/10

---

13. `CALCULATE()`

Explanation: Evaluates an expression in a modified filter context.

Example:

```
TotalSalesFor2019 = CALCULATE(SUM(Sales[Amount]), Sales[Year] = 2019)
```

Rating: Interview Prep: 10/10 | Company Use: 10/10

---

14. `FILTER()`

Explanation: Returns a table that represents a subset of another table or expression.

Example:

```
HighValueSales = FILTER(Sales, Sales[Amount] > 1000)
```

Rating: Interview Prep: 10/10 | Company Use: 10/10

---

15. `ALL()`

Explanation: Removes all filters from a table or column.

Example:

```
AllSales = CALCULATE(SUM(Sales[Amount]), ALL(Sales[Region]))
```

Rating: Interview Prep: 9/10 | Company Use: 10/10

---

16. `ALLEXCEPT()`

Explanation: Removes all filters from the specified table except the ones explicitly mentioned.

Example:

```
SalesWithoutRegionFilter = CALCULATE(SUM(Sales[Amount]), ALLEXCEPT(Sales,
Sales[Region]))
```

Rating: Interview Prep: 8/10 | Company Use: 9/10

---

17. `VALUES()`

Explanation: Returns a one-column table that contains the distinct values from the specified column.

Example:

```
DistinctProducts = VALUES(Sales[ProductID])
```

Rating: Interview Prep: 9/10 | Company Use: 10/10

---

18. `TOPN()`

Explanation: Returns the top N rows of a table.

Example:

```
Top5Customers = TOPN(5, Customers, Customers[TotalSpend], DESC)
```

Rating: Interview Prep: 9/10 | Company Use: 9/10

---

19. `RANKX()`

Explanation: Returns the rank of an expression evaluated for each row in a table.

Example:

```
SalesRank = RANKX(ALL(Sales), Sales[Amount], , DESC)
```

Rating: Interview Prep: 10/10 | Company Use: 10/10

---

20. `DISTINCT()`

Explanation: Returns a table by removing duplicate rows from another table or expression.

Example:

```
UniqueProducts = DISTINCT(Sales[ProductID])
```

Rating: Interview Prep: 9/10 | Company Use: 9/10

---

21. `UNION()`

Explanation: Combines two or more tables by rows, stacking them.

Example:

```
AllSales = UNION(Sales2019, Sales2020)
```

Rating: Interview Prep: 8/10 | Company Use: 8/10

---

22. `INTERSECT()`

Explanation: Returns the intersection of two tables.

Example:

```
CommonProducts = INTERSECT(Products[ProductID], Sales[ProductID])
```

Rating: Interview Prep: 7/10 | Company Use: 7/10

---

23. `EXCEPT()`

Explanation: Returns the rows of one table that do not appear in another table.

Example:

```
NewProducts = EXCEPT(Products[ProductID], Sales[ProductID])
```

Rating: Interview Prep: 7/10 | Company Use: 7/10

---

24. `ADDCOLUMNS()`

Explanation: Adds calculated columns to a table.

Example:

```
SalesWithProfit = ADDCOLUMNS(Sales, "Profit", Sales[Amount] - Sales[Cost])
```

Rating: Interview Prep: 9/10 | Company Use: 9/10

---

25. `SUMMARIZE()`

Explanation: Groups a table by specified columns and calculates aggregations.

Example:

```
SalesSummary = SUMMARIZE(Sales, Sales[Region], "Total Sales", SUM(Sales[Amount]))
```

Rating: Interview Prep: 10/10 | Company Use: 10/10

---

26. `EARLIER()`

Explanation: Returns the value of an expression evaluated in an earlier row context.

Example:

```
SalesCumulative = CALCULATE(SUM(Sales[Amount]), FILTER(Sales, Sales[OrderDate] <=
EARLIER(Sales[OrderDate])))
```

Rating: Interview Prep: 8/10 | Company Use: 8/10

---

27. `ISBLANK()`

Explanation: Checks whether a value is blank (i.e., null).

Example:

```
IsEmptySales = IF(ISBLANK(Sales[Amount]), "No Sales", Sales[Amount])
```

Rating: Interview Prep: 9/10 |