

Data Pipeline with Quality



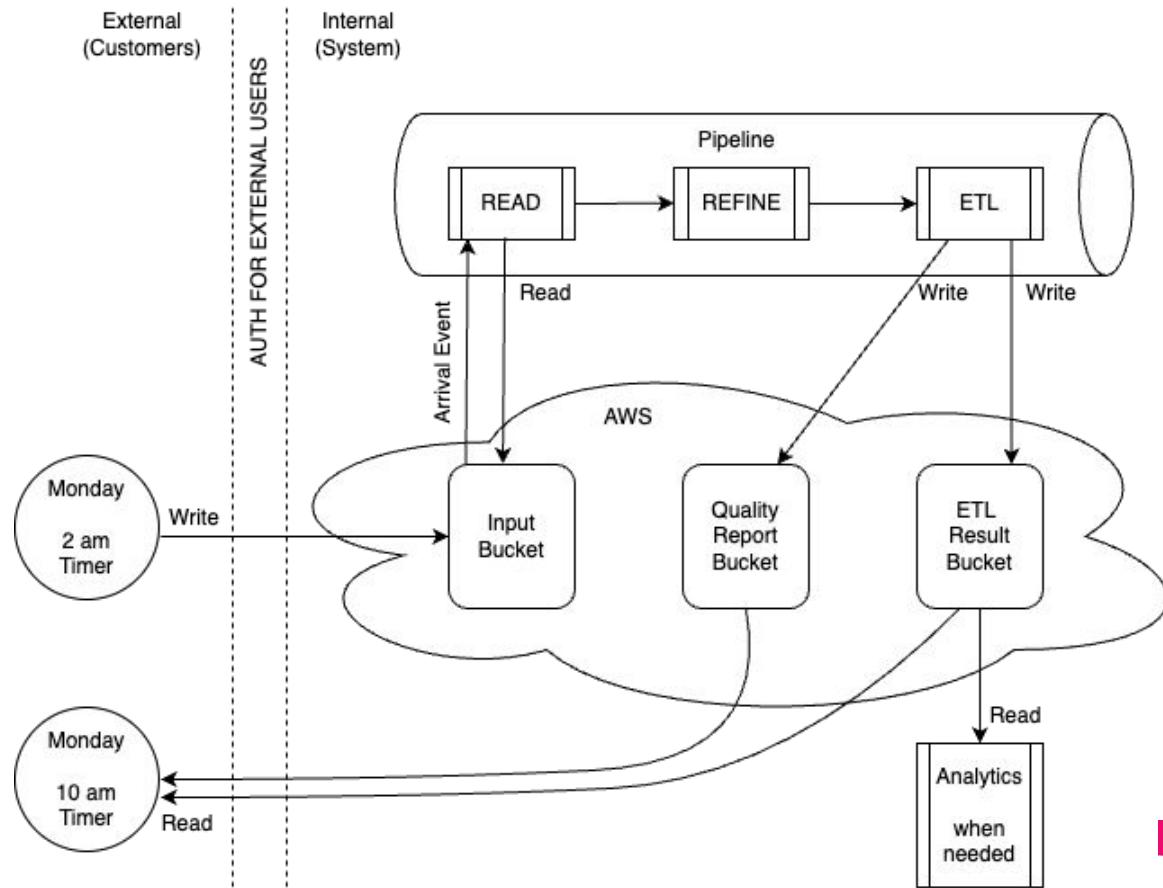
Presented to: Tendo Staff

Presented by: Susan Korgen

May 2, 2024



Pipeline.

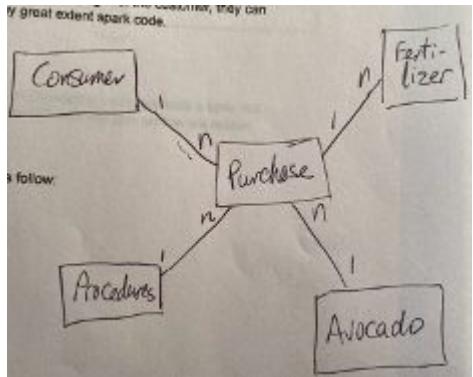


Repo.

The screenshot shows a GitHub repository page for 'susankorgen / de-practice'. The 'Pull requests' tab is selected. A search bar at the top right contains the query 'is:pr is:closed'. Below the search bar, there are buttons for 'Labels 9' and 'Milestones 0', and a green 'New pull request' button. A link to 'Clear current search query, filters, and sorts' is also present. The main area displays a list of 11 closed pull requests, each with a checkbox, a title, and a merge commit message. The pull requests are listed in descending order of creation date.

checkbox	Title	Author	Label	Projects	Milestones	Reviews	Assignee	Sort
<input type="checkbox"/>	#11 add project workbook	susankorgen						
<input type="checkbox"/>	#10 create and output the data quality issue report	susankorgen						
<input type="checkbox"/>	#9 Task2 ETL, run Task1 &/or 2, Task1 unit tests pass	susankorgen						
<input type="checkbox"/>	#8 CSV, Markdown, and HTML versions of the output	susankorgen						
<input type="checkbox"/>	#7 Task 1 done; ETL, ASCII, model classes, file org'n	susankorgen						
<input type="checkbox"/>	#6 output only CSV; use unittest.TestCase, setUp(), tearDown()	susankorgen						
<input type="checkbox"/>	#5 better cleanup after tests, file overwrite works	susankorgen						
<input type="checkbox"/>	#4 able to write output csv file from data	susankorgen						
<input type="checkbox"/>	#3 able to read customer source data csv file	susankorgen						
<input type="checkbox"/>	#2 readme update	susankorgen						
<input type="checkbox"/>	#1 First draft README.md	susankorgen				1		

Day 1.



Day 1

Project Requirements, Success Factors, Constraints ...

Start Workbook

Sprint Plan: 4 Days, 1 Epic, 14 Stories

Scrum

Databricks Knowledge.....

Cloud Knowledge

Spark Knowledge.....

Task 1

Repo and Peer Review

Best Practices

Customer Data Model (Diagram)

Trust but Verify, or: Truth is in the Data

Day 2.

Day 2

Scrum

Databricks Knowledge.....

Task 1

Task 1 ETL Plan

Command Line Demo with tabulate Output.....

Display Options for Lead Business Stakeholder (and Customer)

Task 2

Task 2 ETL Plan

```
def main(args: Array[String]): Unit = {
    // Read the data
    val rawDf = spark.read.json("dbfs:/FileStore/test-data/drugstore/Products.json")
    rawDf.printSchema()
    rawDf.show(10)

    // Infer schema
    val schema = rawDf.schema
    println(schema)
    schema.foreach { field => println(s"Field $field has type ${field.dataType} with options ${field.options}") }

    // Write the output
    rawDf.write.parquet("dbfs:/FileStore/test-data/drugstore/Products.parquet")
}
```

product_id	name	size	price	active_ingredients	price_index	active_ingredients_percent	manufacturing_type
0	11100220402	Relief	10	220	1	90	Inhaler 20000 mg EVERY 4 HOURS PRO
1	11100220403	Relief	12	240	1	92	Inhaler 2 mg DAILY FOR 1 HOURS PRO
2	11100220405	Relief	10	200	1	89	Inhaler 1 mg DAILY AND 2 HOURS PRO
3	11100220406	Relief	12	220	1	91	Inhaler 1 mg DAILY AND 2 HOURS PRO
4	11100220407	Relief	10	210	1	93	Inhaler 20000 mg EVERY 1 HOURS PRO
5	11100220409	Unopened	10	200	1	87	Unopened 1 mg DAILY FOR 1 HOURS PRO
6	11100220409	Unopened	84	260	1	87	Unopened 1 mg DAILY FOR 1 HOURS PRO
7	11100220409	Unopened	84	260	1	87	Unopened 1 mg DAILY FOR 1 HOURS PRO
8	11100220409	Unopened	12	320	1	94	Unopened 2 mg DAILY FOR 1 HOURS PRO
9	11100220409	Non-prescription	12	240	1	94	Non-prescription 2 mg DAILY FOR 1 HOURS PRO
10	11100220409	Non-prescription	25	320	1	94	Non-prescription 5 mg DAILY FOR 1 HOURS PRO
11	11100220409	Non-prescription	25	320	1	94	Non-prescription 10 mg DAILY FOR 1 HOURS PRO
12	11100220409	Relief	24	280	1	95	Relief 1 mg DAILY
13	11100220409	Relief	24	280	1	95	Relief 5 mg DAILY
14	11100220409	Relief	24	280	1	95	Relief 10 mg DAILY
15	11100220409	Relief	24	280	1	95	Relief 20 mg DAILY
16	11100220409	Relief	24	280	1	95	Relief 40 mg DAILY
17	11100220409	Relief	24	280	1	95	Relief 80 mg DAILY
18	11100220409	Relief	24	280	1	95	Relief 100 mg DAILY
19	11100220409	Relief	24	280	1	95	Relief 200 mg DAILY
20	11100220409	Relief	24	280	1	95	Relief 400 mg DAILY
21	11100220409	Relief	24	280	1	95	Relief 800 mg DAILY
22	11100220409	Relief	24	280	1	95	Relief 1600 mg DAILY
23	11100220409	Relief	24	280	1	95	Relief 3200 mg DAILY
24	11100220409	Relief	24	280	1	95	Relief 6400 mg DAILY
25	11100220409	Relief	24	280	1	95	Relief 12800 mg DAILY
26	11100220409	Relief	24	280	1	95	Relief 25600 mg DAILY

Day 3, exploration.

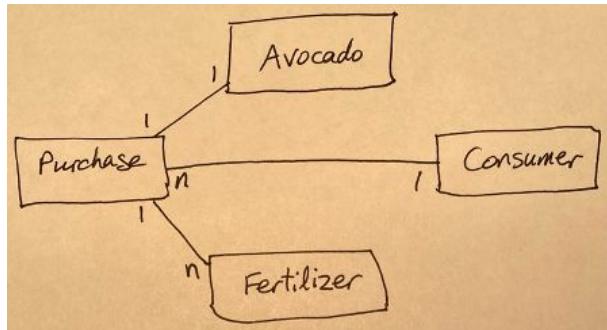


Table Fertilizer:

fertilizerid	NOT NULL PK
consumerid	NOT NULL FK
purchaseid	NOT NULL FK

→ consumer
→ purchase

Table Purchase:

purchaseid	NOT NULL PK
consumerid	NOT NULL FK
avocado_bunch_id	NOT NULL FK

→ consumer
→ avocado
→ purchase

Table Avocado:

avocado_bunch_id	NOT NULL PK
purchaseid	NOT NULL FK
consumerid	NOT NULL FK

→ purchase
→ consumer

Table Consumer:

consumerid	(all values bad)	NOT NULL PK
------------	------------------	-------------

Day 3

Scrum

Task 2

Real-World Data

Data Quality Issues

Data Exploration can reveal DATA DESIGN Quality Issues

Pivot Plan for ETL

ETL Daily Runs can reveal DATA VALUE Quality Issues

Day 3, quality.

Data value quality issues are discoverable during daily processing. From most to least severe:

1. BLOCKER – for the exercise, will report and work around; in real life these must be fixed:
 - a. PK is not unique in table (consumerid) – (for demo: mock values)
 - b. Missing value for a field that is NOT NULLABLE (PKs and FKS) – (for demo: skip row)
2. MISSING – (BLOCKER is also reported for these) Missing PK or FK for our scenario:
 - a. consumer.consumerid
 - b. fertilizer.purchaseid
 - c. fertilizer.consumerid
3. MISSING – Missing value for a data field needed for our scenario, demo substitutes:
 - a. consumer.Sex (for demo: random value)
 - b. consumer.age (for demo: random value)
 - c. avocado.sold_date (for demo: make today)
 - d. avocado.born_date (for demo: make today)
 - e. avocado."ripe index when picked" (for demo: make 0)
 - f. avocado.picked_date (for demo: make today)
4. DATATYPE – will flag and discuss with customer; for demo, will try to adjust data type
 - a. Inconsistent date format values (for demo: strip time from date strings)
 - b. Inappropriate data type for identifier (float for any “id”, should be int or str)
 - c. Date value that is non-empty but not parseable as a date. (for demo: empty)
5. RANGE – will flag; can help customer decide on path forward; for demo, will adjust
 - a. Super old dates (for demo: set to 365)
 - b. Negative ages (for demo: make 0)
 - c. Negative numbers of days (for demo: make 0)
 - d. Negative ripe index (for demo: make 0)
 - e. Ripe index > 10 or < 0 (for demo: adjust to 10 or 0 respectively)
 - f. Dates in the future as of today (for demo: make today)
6. UNEXPECTED – flag it; can help customer decide on path forward; for demo, adjust
 - a. same FK value in 95-100% of rows (consumerid) (for demo: random value)
 - b. non-ASCII char in input but ASCII output is required (for demo: “Invalid”)
 - c. NaN in a float (for demo: 0.0)
 - d. NaN in an int (for demo: 0)
7. NONSTANDARD – will flag; could discuss with customer, but often they can't change
 - a. Spaces in a field name: avocado.”ripe index when picked”
 - b. Capitalized column names: Sex, Race Age in consumer.csv

Day 3

Scrum

Task 2

Real-World Data

Data Quality Issues

Data Exploration can reveal DATA DESIGN Quality Issues

Pivot Plan for ETL

ETL Daily Runs can reveal DATA VALUE Quality Issues

Day 4, quality.

```
data_quality_issue_store: dict = dict(  
{  
    "UNKNOWN": ["UNKNOWN", 0.0, "Invalid issue key in issue report", "ignore"],  
    "BLOCKER-1a": ["BLOCKER-1a", 50.0, "PK is not unique in table", "mock values"],  
    "BLOCKER-1b": ["BLOCKER-1b", 40.0, "Missing value for a field that is NOT NULLABLE (PKs and  
    "MISSING-2a": ["MISSING-2a", 28.0, "Missing value for a field that is NOT NULLABLE (PKs and  
    "MISSING-3a": ["MISSING-3a", 25.0, "Missing value for a field required for demo", "choose random  
    "MISSING-3b": ["MISSING-3b", 23.0, "Missing value for a date field needed for demo", "use today"],  
    "MISSING-3c": ["MISSING-3c", 20.0, "Missing value for a number needed for demo", "use 0"],  
    "DATATYPE-4a": ["DATATYPE-4a", 15.0, "Inconsistent date format values", "strip time off the  
    "DATATYPE-4b": ["DATATYPE-4b", 13.0, "Inappropriate data type for identifier (float)", "convert to string"],  
    "DATATYPE-4c": ["DATATYPE-4c", 10.0, "Date value that is non-empty but not parseable as a date",  
    "RANGE-5a": ["RANGE-5a", 8.0, "Value too low, out of range", "set to minimum in range"],  
    "RANGE-5b": ["RANGE-5b", 7.5, "Value too high, out of range", "set to maximum in range"],  
    "RANGE-5c": ["RANGE-5c", 7.0, "Dates in the future as of today", "set to today"],  
    "UNEXPECTED-6a": ["UNEXPECTED-6a", 5.0, "Same PK or FK value in 95-100% of rows", "mock values"],  
    "UNEXPECTED-6b": ["UNEXPECTED-6b", 4.75, "Non-ASCII char in str but ASCII output is required",  
    "UNEXPECTED-6c": ["UNEXPECTED-6c", 4.50, "NaN in a float", "set to 0.0"],  
    "UNEXPECTED-6z": ["UNEXPECTED-6z", 4.50, "NaN in an int", "set to 0"],  
    "UNEXPECTED-6d": ["UNEXPECTED-6d", 4.25, "Empty number value, could not cast to str", "use empty"],  
    "UNEXPECTED-6e": ["UNEXPECTED-6e", 4.12, "Invalid number value, could not cast to str", "use empty"],  
    "UNEXPECTED-6f": ["UNEXPECTED-6f", 4.0, "Empty string where a date should be", "may use empty"],  
    "UNEXPECTED-6j": ["UNEXPECTED-6j", 3.14159, "Large integer where an int should be", "cast as str"],  
    "UNEXPECTED-6k": ["UNEXPECTED-6k", 2.71828, "Unexpected type (not str, int, or float) where",  
    "UNEXPECTED-6g": ["UNEXPECTED-6g", 0.83, "Invalid str where a date should be", "may use empty"],  
    "UNEXPECTED-6h": ["UNEXPECTED-6h", 0.47, "str where an int should be", "cast as int"],  
    "UNEXPECTED-6i": ["UNEXPECTED-6i", 0.39, "float where an int should be", "cast as int"],  
    "NONSTANDARD-7a": ["NONSTANDARD-7a", 0.2, "Spaces in a field name", "no spaces"],  
    "NONSTANDARD-7b": ["NONSTANDARD-7b", 0.1, "Capitalized column name", "no caps"],  
})
```

Day 4

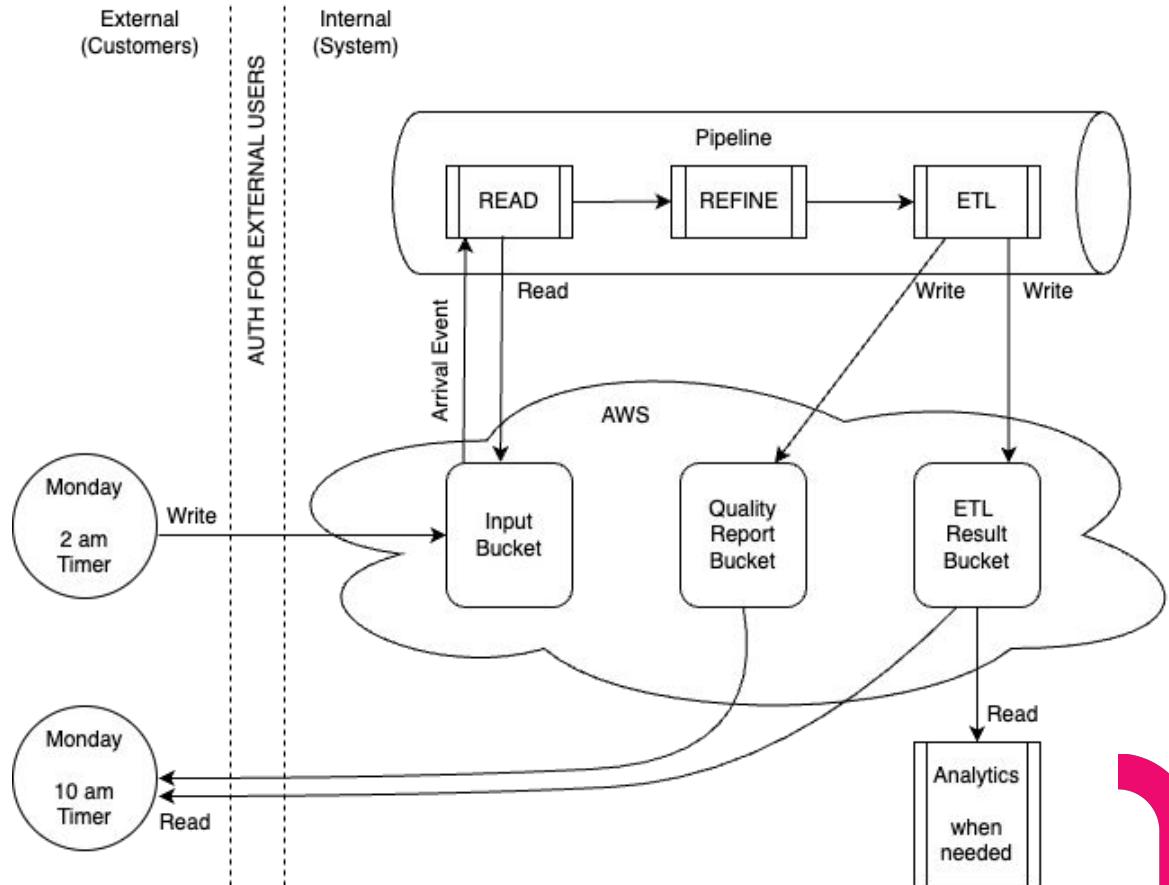
Scrum

- DataFrame, or: Extending a Successful Prototype to Make a Product
- AI Helpers for Work
- Protected Health Information (PHI)

Task 2

- Task 2 ETL Implementation
- Command Line Demo with tabulate output
- Display Options for Lead Business Stakeholder (and Customer)
- Data Pipeline Architecture (Diagram)
- Data Quality Implementation
- Data Quality Issue Report
- AWS S3 Buckets

Day 4, design.



Day 4, report.

	name	severity	description	demo_workaround	pipeline_segment	table_name	field_name
558	BLOCKER-1a	50.00	PK is not unique in table	mock values	transform	consumer	consumerid
562	BLOCKER-1a	50.00	PK is not unique in table	mock values	transform	consumer	consumerid
566	BLOCKER-1a	50.00	PK is not unique in table	mock values	transform	consumer	consumerid
569	BLOCKER-1a	50.00	PK is not unique in table	mock values	transform	consumer	consumerid
573	BLOCKER-1a	50.00	PK is not unique in table	mock values	transform	consumer	consumerid
577	BLOCKER-1a	50.00	PK is not unique in table	mock values	transform	consumer	consumerid
581	BLOCKER-1a	50.00	PK is not unique in table	mock values	transform	consumer	consumerid
585	BLOCKER-1a	50.00	PK is not unique in table	mock values	transform	consumer	consumerid
588	BLOCKER-1a	50.00	PK is not unique in table	mock values	transform	consumer	consumerid
589	MISSING-3a	25.00	Missing value for a field required for demo	choose random row from real data	transform	consumer	consumerid
593	MISSING-3a	25.00	Missing value for a field required for demo	choose random row from real data	transform	consumer	consumerid
597	MISSING-3a	25.00	Missing value for a field required for demo	choose random row from real data	transform	consumer	consumerid
600	MISSING-3a	25.00	Missing value for a field required for demo	choose random row from real data	transform	consumer	consumerid
674	MISSING-3a	25.00	Missing value for a field required for demo	choose random row from real data	transform	consumer	consumerid
678	MISSING-3a	25.00	Missing value for a field required for demo	choose random row from real data	transform	consumer	consumerid
582	MISSING-3a	25.00	Missing value for a field required for demo	choose random row from real data	transform	consumer	consumerid
586	MISSING-3a	25.00	Missing value for a field required for demo	choose random row from real data	transform	consumer	consumerid
589	MISSING-3a	25.00	Missing value for a field required for demo	choose random row from real data	transform	consumer	consumerid
572	RANGE-5a	8.00	Value too low, out of range	set to minimum in range	transform	consumer	Age
580	RANGE-5b	8.00	Value too low, out of range	set to minimum in range	transform	consumer	Age
561	RANGE-5b	7.50	Value too high, out of range	set to maximum in range	transform	consumer	Age
565	RANGE-5b	7.50	Value too high, out of range	set to maximum in range	transform	consumer	Age
576	RANGE-5b	7.50	Value too high, out of range	set to maximum in range	transform	consumer	Age
584	RANGE-5b	7.50	Value too high, out of range	set to maximum in range	transform	consumer	Age
557	UNEXPECTED-6b	4.75	Non-ASCII char in str but ASCII output is required	use 'other'	efine_input	fertilize	type
1	UNEXPECTED-6c	0.39	float where an int should be	cast as int	efine_input	consumer	consumerid
2	UNEXPECTED-6c	0.39	float where an int should be	cast as int	efine_input	consumer	consumerid
3	UNEXPECTED-6c	0.39	float where an int should be	cast as int	efine_input	consumer	consumerid
4	UNEXPECTED-6c	0.39	float where an int should be	cast as int	efine_input	consumer	consumerid
5	UNEXPECTED-6c	0.39	float where an int should be	cast as int	efine_input	consumer	consumerid
6	UNEXPECTED-6c	0.39	float where an int should be	cast as int	efine_input	consumer	consumerid
7	UNEXPECTED-6c	0.39	float where an int should be	cast as int	efine_input	consumer	consumerid
8	UNEXPECTED-6c	0.39	float where an int should be	cast as int	efine_input	consumer	Age
9	UNEXPECTED-6c	0.39	float where an int should be	cast as int	efine_input	consumer	Age
10	UNEXPECTED-6c	0.39	float where an int should be	cast as int	efine_input	consumer	Age
11	UNEXPECTED-6c	0.39	float where an int should be	cast as int	efine_input	consumer	Age
12	UNEXPECTED-6c	0.39	float where an int should be	cast as int	efine_input	consumer	Age
13	UNEXPECTED-6c	0.39	float where an int should be	cast as int	efine_input	consumer	Age
14	UNEXPECTED-6c	0.39	float where an int should be	cast as int	efine_input	consumer	Age
15	UNEXPECTED-6c	0.39	float where an int should be	cast as int	efine_input	purchase	purchaseid
16	UNEXPECTED-6c	0.39	float where an int should be	cast as int	efine_input	purchase	purchaseid
17	UNEXPECTED-6c	0.39	float where an int should be	cast as int	efine_input	purchase	purchaseid

Day 4

Scrum

- DataFrame, or: Extending a Successful Prototype to Make a Product
- AI Helpers for Work
- Protected Health Information (PHI)

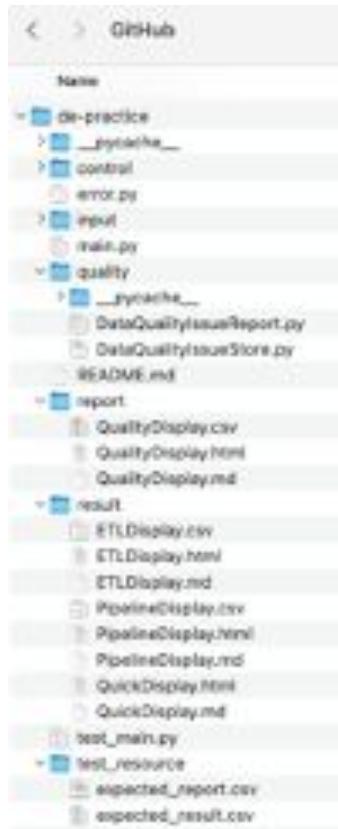
Task 2

- Task 2 ETL Implementation
- Command Line Demo with tabulate output
- Display Options for Lead Business Stakeholder (and Customer)
- Data Pipeline Architecture (Diagram)
- Data Quality Implementation
- Data Quality Issue Report
- AWS S3 Buckets

Day 4, buckets.

Using the local file system as if folders represent buckets. Not shown: we must write each different customer's output to different locations to protect against write collisions and privacy leaks.

1. Pipeline Output is in **result**
2. Data Quality Output in **report**



Days 1-4, repo.

<https://github.com/susankorgen/de-practice>

Questions?



Some topics:

- Stakeholder communication
- Agile for project management
- Coding best practices
- Real-world data
- In-real-life moments
- Demo (Task 1) vs. product (Task 2)
- Extending with new features





Ongoing innovation with Tendo.

