Cleaning CDDB

This notebook is an annotated walkthrough of cleaning the Compact Disc Database (CDDB) dataset.

In [ ]:
```
import logging

import pandas as pd
import pandera as pa

import clean_cddb
from clean_cddb.utils import get_failure_cases_summary_as_formatted_table, get_check_func_descriptions

def df_to_var(df, var_name):
    globals()[var_name] = df
    return df

pd.set_option("display.max_rows", 1000)
pd.set_option("display.max_columns", None)
pd.set_option("display.max_colwidth", None)

logging.basicConfig(
    level=logging.INFO,
    format="%(asctime)s - %(process)d - %(levelname)s - %(message)s",
)
filepath = "../data/input/cddb.tsv"
source_df = pd.read_csv(filepath, sep="\t", dtype="str", encoding='latin1')
```

Apply validation checks (pandera schema) and review failure cases

In [ ]:
```
try:
    validated_df = clean_cddb.schema(source_df, lazy=True)
    logging.info("Validation success. No failure cases detected.")
except pa.errors.SchemaErrors as err:
    logging.info("Validation failure. Failure cases detected.")
    logging.debug(err)
    failure_cases_df = err.failure_cases

failure_cases_df = failure_cases_df.pipe(get_check_func_descriptions, clean_cddb.schema)
```

```
2023-07-30 15:25:48,509 - 87842 - INFO - Validation failure. Failure cases detected.
```

`failure_cases_df`

- The `failure_cases_df` shows the name of the column, the check, failure case (example), and row index position of the failure case in the original data frame.
- The index can support bulk operations such as joining and querying the original dataframe for failure cases or rejecting rows in the set of failure case indices.

In [ ]:
```
(
    failure_cases_df
    .loc[
        :, ["schema_context", "column", "check", "failure_case", "index"]
    ]
    .sample(10, random_state=1)
)
```

Out[ ]:

	schema_context	column	check	failure_case	index
5567	Column	title	Check for *possibly* invalid symbols.	UltrasÃ Â³nica	9646
1145	Column	artist	Check for *possibly* invalid symbols.	BjÃ Â¶rk	6862
5766	Column	id	Check that the length of 'id' is 6 characters.	3931	9397
5684	Column	id	Check that the length of 'id' is 6 characters.	10938	8938
1302	Column	artist	Check for *possibly* invalid symbols.	Gilbert MontagnÃ Â©	9211
4822	Column	title	not_nullable	NaN	7705
4431	Column	genre	not_nullable	NaN	8596
4949	Column	title	Check for *possibly* invalid symbols.	Gympa PÃ Â¥	1402
3222	Column	genre	not_nullable	NaN	5074
5216	Column	title	Check for *possibly* invalid symbols.	Ã â Ã Â¥Ã Â¢Ã Â®Ã Â·Ã Â³Ã Â Ã Ë Ã ï Ã Â»Ã Â±Ã Â¼	4708

Summary of failure cases

Here we see aggregated counts of the number of failure cases for each validation check.

In [ ]:
```
failure_cases_summary = (
    failure_cases_df.groupby(["column", "check"], as_index=False)
    .size()
    .sort_values(by=["column", "check"])
    .rename(columns={"size": "counts"})
)

failure_cases_summary
```

Out[ ]:

	column	check	counts
0	artist	Check for *possibly* invalid symbols.	639
1	artist	Check for invalid artist values.	697
2	artist	not_nullable	1
3	category	Check for invalid categories.	89
4	genre	Check for invalid genres.	1
5	genre	not_nullable	3388
6	id	Check that the length of 'id' is 6 characters.	477
7	title	Check for *possibly* invalid symbols.	747
8	title	not_nullable	8
9	year	Check if year is numeric.	28
10	year	Check that year is between 1950 and 2030.	56
11	year	dtype('Int32')	1

We also have a helper utility function to display the source code along side each check function name.

In [ ]:

```
# Report summary counts
failure_cases_summary_table = get_failure_cases_summary_as_formatted_table(failure_cases_df)

print(failure_cases_summary_table)
```

column	check	counts	check_source_code

artist	Check for *possibly* invalid symbols.	639	def check_col_has_valid_characters(x: Any) -> bool:
			    """Check for *possibly* invalid symbols."""
			
			    # consider NaNs and floats to be invalid
			    if not isinstance(x, str):
			        return False
			
			    invalid_symbols = set(
			        "\^¤¦©¬®¯°±²³´µ¶¸¹º»¼½¾¿ÀÃÄÅÆÇÌÖÖÜàåçïð÷øùû˜▢↯€中俊劇四団季雅
			    ) # noqa
			    for char in x:
			        if char in invalid_symbols:
			            return False
			
			    chinese_pattern = re.compile(r"[\u4e00-\u9fff]") # CJK Unified Ideographs
			    japanese_pattern = re.compile(
			        r"[\u3040-\u30ff\u31f0-\u31ff\u3200-\u9faf]"
			    ) # Hiragana, Katakana, CJK Unified Ideographs Extension A
			    has_chinese = bool(chinese_pattern.search(x))
			    has_japanese = bool(japanese_pattern.search(x))
			
			    if has_chinese or has_japanese:
			        return False
			
			    return True

artist	Check for invalid artist values.	697	def check_artist_is_valid(x: Any) -> bool:
			    """Check for invalid artist values."""
			
			    various_artist_pattern = r"\b(various|various artist(s)?|var)\b"
			    try:
			        # match on variations of "various", "various artist",
			        # "various artists", and "var"; case-insensitive
			        # also match 2 or more question marks like "????" or "?? ??"
			        if _match := re.search(various_artist_pattern, re.escape(x), re.IGNORECASE):
			            if x != "Various":
			                return False
			
			        if "??" in x:
			            return False
			    except Exception:
			        return False
			
			    return True

category	Check for invalid categories.	89	def check_category_is_valid(x: Any) -> bool:
			    """Check for invalid categories."""
			
			    if x in [
			        "blues",
			        "classical",
			        "country",
			        "folk",
			        "jazz",
			        "misc",
			        "newage",
			        "reggae",
			        "rock",
			        "soundtrack",
			        "N/A",
			        # "REJECT_ROW",
			    ]:
			        return True
			    else:
			        return False

genre	Check for invalid genres.	1	def check_genre_is_valid(x: Any) -> bool:
			    """Check for invalid genres."""
			    try:
			        if "--" in x:
			            return False
			    except Exception:
			        return False
			    else:
			        return True

id	Check that the length of 'id' is 6 characters.	477	lambda x: len(x)==6

title	Check for *possibly* invalid symbols.	747	def check_col_has_valid_characters(x: Any) -> bool:
			    """Check for *possibly* invalid symbols."""
			
			    # consider NaNs and floats to be invalid
			    if not isinstance(x, str):
			        return False
			
			    invalid_symbols = set(
			        "\^¤¦©¬®¯°±²³´µ¶¸¹º»¼½¾¿ÀÃÄÅÆÇÌÖÖÜàåçïð÷øùû˜▢↯€中俊劇四団季雅
			    ) # noqa
			    for char in x:
			        if char in invalid_symbols:
			            return False
			
			    chinese_pattern = re.compile(r"[\u4e00-\u9fff]") # CJK Unified Ideographs
			    japanese_pattern = re.compile(
			        r"[\u3040-\u30ff\u31f0-\u31ff\u3200-\u9faf]"
			    ) # Hiragana, Katakana, CJK Unified Ideographs Extension A
			    has_chinese = bool(chinese_pattern.search(x))
			    has_japanese = bool(japanese_pattern.search(x))
			
			    if has_chinese or has_japanese:
			        return False
			
			    return True

year	Check if year is numeric.	28	def check_year_is_numeric(x: Any) -> bool:
			    """Check if year is numeric."""
			
			    try:
			        if not str(int(x)).isnumeric():

				return False
				except Exception:
				return False
				return True
year	Check that year is between 1950 and 2030.		56	def check_year_range_is_valid(x: Any) -> bool:
				"""Check that year is between 1950 and 2030."""
				try:
				int(x)
				except Exception:
				return False
				if int(x) > 1950 and int(x) < 2030:
				return True
				else:
				return False

Cleaning step

We can use the same checks from the pandera validation schema to trigger cleaning actions such as:

- do nothing / ignore the value
- transform the value; e.g., replace value with a substitute (e.g., 'N/A')
- or reject the entire record

Here we apply several cleaning functions on the source_df via .pipe(Callable).

- Each function takes a dataframe and returns a dataframe, so we can chain together the cleaning operations like so.
- Later, we will
  1. compare `source_df` and `clean_df` as a before/after check
  2. re-apply our validation checks (pandera schema) to the new `clean_df` to verify that our transformations improved our data quality

In [ ]:
```python
from clean_cddb.utils import log_df_change

clean_df = (
source_df

    .pipe(df_to_var, '_df_before').pipe(clean_cddb.clean_df_standardize_various_artists)
    .pipe(log_df_change, before_df=_df_before, operation_label="Cleaning with 'clean_cddb.clean_df_try_to_fix_encoding_errors' procedure")

    .pipe(df_to_var, '_df_before').pipe(clean_cddb.clean_df_try_to_fix_encoding_errors, "artist")
    .pipe(log_df_change, before_df=_df_before, operation_label="Cleaning with 'clean_cddb.clean_df_try_to_fix_encoding_errors' procedure")
    .pipe(df_to_var, 'clean_df_artist_transforms_only')

    .pipe(df_to_var, '_df_before').pipe(clean_cddb.clean_df_invalid_symbols)
    .pipe(log_df_change, before_df=_df_before, operation_label="Cleaning with 'clean_cddb.clean_df_invalid_symbols' procedure")

    .pipe(df_to_var, '_df_before').pipe(clean_cddb.clean_df_invalid_categories)
    .pipe(log_df_change, before_df=_df_before, operation_label="Cleaning with 'clean_cddb.clean_df_invalid_categories' procedure")

    .pipe(df_to_var, '_df_before').pipe(clean_cddb.clean_df_id_zero_padding)
    .pipe(log_df_change, before_df=_df_before, operation_label="Cleaning with 'clean_cddb.clean_df_id_zero_padding' procedure")

    .pipe(df_to_var, '_df_before').pipe(clean_cddb.clean_df_genre_invalid)
    .pipe(log_df_change, before_df=_df_before, operation_label="Cleaning with 'clean_cddb.clean_df_genre_invalid' procedure")

    .pipe(df_to_var, '_df_before').pipe(clean_cddb.clean_df_year)
    .pipe(log_df_change, before_df=_df_before, operation_label="Cleaning with 'clean_cddb.clean_df_year' procedure")

    .pipe(df_to_var, '_df_before').pipe(clean_cddb.clean_df_title)
    .pipe(log_df_change, before_df=_df_before, operation_label="Cleaning with 'clean_cddb.clean_df_title' procedure")

    .pipe(df_to_var, '_df_before').pipe(clean_cddb.clean_df_genre_coalesce_with_category)
    .pipe(log_df_change, before_df=_df_before, operation_label="Cleaning with 'clean_cddb.clean_df_genre_coalesce_with_category' procedure")

    # Save an intermediate dataframe prior to dropping records
    # so we can compare with source_df later
    .pipe(df_to_var, 'clean_df_before_drops')

    # Drop rows with "REJECT_ROW*" prefix
    .query("~id.str.contains('REJECT_ROW')")
    .drop(columns=['merged_values'])
)
```

```
2023-07-30 15:25:48,567 - 87842 - INFO - Cleaning operation
operation_label: Cleaning with 'clean_cddb.clean_df_try_to_fix_encoding_errors' procedure
cleaning operation_label: Cleaning with 'clean_cddb.clean_df_try_to_fix_encoding_errors' procedure
Number of rows affected: 703
Columns affected: {'artist'}
Examples:
|      | ('artist', 'before')   | ('artist', 'after')   |
|-----:|:-----------------------|:----------------------|
| 7375 | Various Artists        | Various               |
| 6256 | Various Artists        | Various               |
| 7495 | Various Artists        | Various               |
| 8171 | Various Artist         | Various               |
| 7012 | Various Artists        | Various               |
==========================================================================================
==========================================================================================

2023-07-30 15:25:48,622 - 87842 - INFO - Cleaning operation
operation_label: Cleaning with 'clean_cddb.clean_df_try_to_fix_encoding_errors' procedure
cleaning operation_label: Cleaning with 'clean_cddb.clean_df_try_to_fix_encoding_errors' procedure
Number of rows affected: 292
Columns affected: {'artist'}
Examples:
|      | ('artist', 'before')   | ('artist', 'after')   |
|-----:|:-----------------------|:----------------------|
| 7679 | Noir DÃ Æ Ã Â©sir                       | Noir Désir            |
| 9401 | Los CaÃ Â±os                        | Los Caños             |
| 6998 | Olle LjungstrÃ Â¶m                       | Olle Ljungström       |
| 4711 | Herbert GrÃ Â¶nemeyer                       | Herbert Grönemeyer    |
| 3004 | LeÃ Â¦ther Strip                        | Leæther Strip         |
==========================================================================================
==========================================================================================

2023-07-30 15:25:48,864 - 87842 - INFO - Cleaning operation
operation_label: Cleaning with 'clean_cddb.clean_df_invalid_symbols' procedure
cleaning operation_label: Cleaning with 'clean_cddb.clean_df_invalid_symbols' procedure
Number of rows affected: 346
Columns affected: {'artist', 'tracks', 'genre', 'title', 'id', 'category', 'year', 'merged_values'}
Examples:
|      | ('artist', 'before')   | ('artist', 'after')         | ('category', 'before')   | ('category', 'after')       | ('genre', 'before')   | ('genre', 'after')
| ('title', 'before')   | ('title', 'after')           | ('tracks', 'before')
| ('tracks', 'after')         | ('year', 'before')   | ('year', 'after')           |   ('id', 'before') | ('id', 'after')             | ('merged_values', 'before')   |
('merged_values', 'after')   |
|-----:|:-----------------------|:----------------------------|:-------------------------|:----------------------------|:----------------------|:-----------------------
-----|:---------------------|:---------------------------|:-------------------------------------------------------------------------------------------------------------------
---------------------------------------------------------------------------------------------------------------------------------------------------------------
---------------------------------------------------------------------------------------------------------------------|:---------------------------|:---------------------
|:---------------------------|-------------------:|:----------------------------|:--------------------------------|:-----------------------------|
| 9296 | Ã Â±Ã Â¨ Ã Â´Ã Â«Ã Ë Ã Â«                        | REJECT_ROW - invalid artist | soundtrack               | REJECT_ROW - invalid artist | Soundtrack
| REJECT_ROW - invalid artist | Ã Â¹Ã Å Ã Â¼Ã ÂºÃ Â°Ã Â¼Ã Â¿Ã Â· Ã ÂµÃ Â¿Ã Â¹Ã Â°Ã Â¿Ã Â¸                        | REJECT_ROW - invalid artist | Ã Â±Ã Â¨ Ã Â´Ã Â«Ã Ë Ã Â« /
Ã Â»Ã Â§Ã Â¶Ã Â»Ã â ¡Ã ï  Ã Â´Ã â   Ã Â³Ã Â¯Ã Â¿Ã Â¡ | Ã Â±Ã Â¨ Ã Â´Ã Â«Ã Ë Ã Â« / Ã Â½Ã Æ Ã Â³Ã Â´Ã Â½Ã Æ Ã Â½Ã Âº | Ã Â±Ã Â¨ Ã Â´Ã Â«Ã Ë Ã Â« / She is ... one | Ã Â±Ã Â¨
Ã Â´Ã Â«Ã Ë Ã Â« / Ã Â¾Ã â  Ã â ¬Ã Å¡Ã Â¼Ã â   Ã Â¼Ã Â¼Ã Â»Ã Â³ | Ã Â±Ã Â¨ Ã Â¾Ã Â§Ã Ë Ã Â± / Ã Â°Ã ÂºÃ Æ Ã ÂµÃ Â°Ã Å¡Ã Â´Ã â   Ã Â±Ã Â¦ | Ã Â±Ã Â¨ Ã Â¾Ã Â§Ã Ë Ã Â± / Ã Â»Ã Â
§Ã Â¶Ã Â»Ã â ¬Ã â ¡ Ã â ¬Ã Å½Ã Â»Ã Â§( (Edward Elgar) | Ã Â±Ã Â¨ Ã Â´Ã Â«Ã Ë Ã Â« / She is ... two | Ã Â±Ã Â¨ Ã Â´Ã Â«Ã Ë Ã Â« / Ã Â±Ã â  Ã Â¸Ã Â²Ã Â¼Ã â   Ã â ¬Ã Â¸Ã Â·Ã Â½
| Ã Â±Ã Â¨ Ã Â´Ã Â«Ã Ë Ã Â« / Ã Â¹Ã Å Ã Â¼Ã ÂºÃ Â°Ã Â¼Ã Â¿Ã Â· Ã ÂµÃ Â¿Ã Â¹Ã Â°Ã Â¿Ã Â¸ | Ã Â±Ã Â¨ Ã Â´Ã Â«Ã Ë Ã Â« / Ã Â±Ã â ºÃ Â¿Ã Â¡Ã Â°Ã â   | Ã Â±Ã Â¨ Ã Â´Ã Â«Ã Ë Ã Â«
/ Ã ï  Ã Å¸Ã Â¾Ã â  Ã Â¼Ã Â±Ã Â¿Ã Â¡Ã Â¼Ã Â | Ã Â±Ã Â¨ Ã Â¾Ã Â§Ã Ë Ã Â± / The tea time | Ã Â±Ã Â¨ Ã Â´Ã Â«Ã Ë Ã Â« / Ã ÂµÃ Â¿Ã Â¹Ã Â°Ã ÂºÃ Â´Ã Â¿Ã Â¸ | Ã Â±Ã Â¨ Ã Â
´Ã Â«Ã Ë Ã Â« / Ã Â»Ã Â§Ã Â¶Ã Â»Ã â ¡Ã ï  Ã Â´Ã â   Ã Â³Ã Â¯Ã Â¿Ã Â¡ (without vocal) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| REJECT_ROW - invalid artist | 2000                 | REJECT_ROW - invalid artist |             109743 | REJECT_ROW - invalid artist |                                 |
REJECT_ROW - invalid artist  |
| 1549 | Ã Â¦Ã Â¸Ã¢â ¬Â¦Ã Â¨Ã Â²Ã Â´                        | REJECT_ROW - invalid artist | misc                     | REJECT_ROW - invalid artist | Pop
| REJECT_ROW - invalid artist | I'll be there         | REJECT_ROW - invalid artist | ___y  1 | ___y  2 | ___y  3 | ___y  4 | ___y  5 | ___y  6 | ___y  7 | ___y  8 | __
_y  9 | ___y  10 | ___y  11 | ___y  12 | ___y  13 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | REJECT_ROW - invalid artist |
2001                 | REJECT_ROW - invalid artist |             101595 | REJECT_ROW - invalid artist |                                 | REJECT_ROW - invalid artist  |
| 9038 | Ã Â®eÃ Â¯Ã ÂªÃ Â¨Ã Â                        | REJECT_ROW - invalid artist | soundtrack               | REJECT_ROW - invalid artist |                       | REJ
ECT_ROW - invalid artist | Ã Â½Ã â  Ã Â¨Ã â  Ã Â·RÃ Â§Ã Å¡                        | REJECT_ROW - invalid artist | Ã Â¬Ã â ºÃ Â·Ã ÂÃ Â¨Ã Â«Ã Â¿E | NOW AND FOREVER | GOODBYE
| Ã Â¬Ã Â¯Ã ÂªÃ Â¡Ã Â½Ã Â¹Ã Â½Ã Âº | Ã Â·qÃ Â½Ã ï  Ã Â¯dÃ Â¨B | Ã Â½Ã â  Ã Â¨Ã â  Ã Â·RÃ Â§Ã Å¡ | Ã Â¦nÃ Â¤kÃ Â«Ã â   | Ã Â¬Ã Â°Ã Â§AÃ Â¸UÃ Â·Ã Â³ | Ã Â·RÃ Â¤Ã Â£Ã Â·R | Ã Â
¼Ã Â´Ã Â°w | Ã Â³Ã Å Ã Â¦nÃ Â®Ã â °Ã Â¥Ã Âº |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| REJECT_ROW - invalid artist |                      | REJECT_ROW - invalid artist |             109480 | REJECT_ROW - invalid artist |                                 |
REJECT_ROW - invalid artist  |
| 4592 | Ã Â¥Ã Â®Ã¢â ¬Â°Ã Â¨Ã Â¥Ã Â¿Ã Â£?Ã Â²Ã Â£Ã¢â ¬Å¡?Ã Â£?Ã¢â ¬Å                        | REJECT_ROW - invalid artist | rock                     | REJECT_ROW - inval
id artist | JPop                  | REJECT_ROW - invalid artist | True Love             | REJECT_ROW - invalid artist | True Love  EUROBEAT Ver. | Reality>Eternity  EUR
OBEAT Ver. | True Love  DISCO Ver. | Reality>Eternity  DISCO Ver. | True Love  J-POP Ver. | Reality>Eternity  J-POP Ver. | True Love  EUROBEAT Ver. Ã Â¯Ã Â¼Ã â  KARAOKE
Ã Â¯Ã Â¼Ã¢â ¬Â° | Reality>Eternity  EUROBEAT Ver. Ã Â¯Ã Â¼Ã â  KARAOKEÃ Â¯Ã Â¼Ã¢â ¬Â° | True Love  DISCO Ver. Ã Â¯Ã Â¼Ã â  KARAOKEÃ Â¯Ã Â¼Ã¢â ¬Â° | Reality>Eternity  DISC
O Ver. Ã Â¯Ã Â¼Ã â  KARAOKEÃ Â¯Ã Â¼Ã¢â ¬Â° | True Love  J-POP Ver. Ã Â¯Ã Â¼Ã â  KARAOKEÃ Â¯Ã Â¼Ã¢â ¬Â° | Reality>Eternity  J-POP Ver. Ã Â¯Ã Â¼Ã â  KARAOKEÃ Â¯Ã Â¼Ã¢â ¬Â° |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| REJECT_ROW - invalid artist | 2000                 | REJECT_ROW - invalid artist |             104751 | REJECT_ROW - invalid artist |                                 |
REJECT_ROW - invalid artist  |
|   47 | Ã Â¥Ã Â±Ã Â±Ã Â¥Ã Â´Ã Â½Ã Â£?Ã Â¾Ã Â£?Ã¢â ¬Â¢Ã Â£Ã¢â ¬Å¡Ã â  Ã Â£?Ã¢â ¬â                        | REJECT_ROW - invalid artist | data                     | REJE
CT_ROW - invalid artist | JPop                  | REJECT_ROW - invalid artist | HOME                  | REJECT_ROW - invalid artist | Fat Mama | Ã Â£Ã¢â ¬Å¡Ã Â¢Ã Â
£Ã â  Ã¢â ¬Â°Ã Â£Ã â  Ã Â¬Ã Â£Ã â  Ã Â Ã Â£Ã â  Ã ÂªÃ Â£Ã â  Ã Â³ | Ã Â£Ã¢â ¬Å¡Ã Â»Ã Â£Ã â  Ã ÂÃ Â£Ã â  Ã Âª | Ã Â£Ã â  Ã¢â  Â¢Ã Â£Ã â  Ã Â³Ã Â£Ã¢â ¬Å¡Ã Â¸Ã Â£Ã â  Ã Â£Ã Â
£Ã â  Ã Â¸Ã Â£Ã â  Ã Â³ | Ã Â£Ã¢â ¬Å¡Ã Â¹Ã Â£Ã¢â ¬Å¡Ã Â¯Ã Â£Ã â  Ã ÂªÃ Â£Ã â  Ã Â¼Ã Â£Ã â  Ã Â¸Ã Â£Ã â  Ã Â³'97 | Ã Â¥??Ã Â¥Ã¢â ¬Â°?Ã Â£?Ã Â®Ã Â£?Ã ÂªÃ Â£?Ã¢â ¬Å¾Ã Â©Ã Â³Ã Â¥
| Ã Â£Ã¢â ¬Å¡Ã Â³Ã Â£Ã â  Ã Â¡Ã Â£Ã â  Ã Â«Ã Â£Ã â  Ã¢â ¬Â¹Ã Â£Ã¢â ¬Å¡Ã Â¯Ã Â£Ã¢â ¬Å¡Ã Â¹Ã Â£?Ã Â®Ã Â¥?Ã Âµ | Ã Â¥Ã â  Ã¢â ¬Â¢Ã Â£Ã¢â ¬Å¡Ã¢â ¬Â°Ã Â£?Ã Â®Ã Â§Ã¢â ¬Â¦Ã Â©Ã Â¦
Ã¢â ¬Å¡Ã Â© | Ã Â¦Ã Å Ã Â¼Ã Â¤Ã Â¼Ã¢â ¬Ë Ã Â£?Ã Â¿ | One more time,One more chance | Ã Â£Ã â  Ã Â¤Ã Â£Ã¢â ¬Å¡Ã ÂµÃ Â§Ã¢â ¬ï  Ã Â·Ã Â£?Ã Â®Ã Â¥Ã Â¤Ã Â¢ | HOME |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| REJECT_ROW - invalid artist | 1997                 | REJECT_ROW - invalid artist |             100051 | REJECT_ROW - invalid artist |                                 |
REJECT_ROW - invalid artist  |
==========================================================================================
==========================================================================================

2023-07-30 15:25:48,878 - 87842 - INFO - Cleaning operation
operation_label: Cleaning with 'clean_cddb.clean_df_invalid_categories' procedure
cleaning operation_label: Cleaning with 'clean_cddb.clean_df_invalid_categories' procedure
Number of rows affected: 420
Columns affected: {'category'}
Examples:
|      | ('category', 'before')      | ('category', 'after')   |
|-----:|:----------------------------|:------------------------|
| 1688 | REJECT_ROW - invalid artist | N/A                     |
| 8031 | REJECT_ROW - invalid artist | N/A                     |
|   22 | data                        | N/A                     |
|  685 | REJECT_ROW - invalid artist | N/A                     |
| 1963 | REJECT_ROW - invalid artist | N/A                     |
==========================================================================================
==========================================================================================

2023-07-30 15:25:48,890 - 87842 - INFO - Cleaning operation
operation_label: Cleaning with 'clean_cddb.clean_df_id_zero_padding' procedure
cleaning operation_label: Cleaning with 'clean_cddb.clean_df_id_zero_padding' procedure
Number of rows affected: 461
```

```
Columns affected: {'id'}
Examples:
|      |   ('id', 'before') |   ('id', 'after') |
|-----:|-------------------:|------------------:|
| 9631 |               8063 |            008063 |
| 9452 |               5025 |            005025 |
| 9432 |               4631 |            004631 |
| 9373 |               3125 |            003125 |
| 9661 |               8445 |            008445 |
================================================================================================
================================================================================================

2023-07-30 15:25:49,118 - 87842 - INFO - Cleaning operation
operation_label: Cleaning with 'clean_cddb.clean_df_genre_invalid' procedure
cleaning operation_label: Cleaning with 'clean_cddb.clean_df_genre_invalid' procedure
Number of rows affected: 3338
Columns affected: {'artist', 'tracks', 'genre', 'title', 'id', 'category', 'year', 'merged_values'}
Examples:
|      | ('artist', 'before')   | ('artist', 'after')   | ('category', 'before')   | ('category', 'after')   | ('genre', 'before')   | ('genre', 'after')   | ('title',
'before')   | ('title', 'after')   | ('tracks', 'before')   | ('tracks', 'after')   | ('year', 'before')   | ('year', 'after')   | ('id', 'before')   | ('id', 'after')
| ('merged_values', 'before')   | ('merged_values', 'after')   |
|-----:|:----------------------|:----------------------|:-------------------------|:-------------------------|:----------------------|:---------------------|:----------
-----------|:---------------------|:----------------------|:----------------------|:---------------------|:-------------------|:-------------------|:---------------
--|:-----------------------------|:-----------------------------|
| 6741 |                       |                       |                          |                          |                       |                      | N/A                |
|                       |                      |                       |                          |                          |                       |                      |                    |
|                       |
| 1683 |                       |                       |                          |                          |                       |                      | N/A                |
|                       |                      |                       |                          |                          |                       |                      |                    |
|                       |
| 2982 |                       |                       |                          |                          |                       |                      | N/A                |
|                       |                      |                       |                          |                          |                       |                      |                    |
|                       |
| 3537 |                       |                       |                          |                          |                       |                      | N/A                |
|                       |                      |                       |                          |                          |                       |                      |                    |
|                       |
| 5257 |                       |                       |                          |                          |                       |                      | N/A                |
|                       |                      |                       |                          |                          |                       |                      |                    |
|                       |
================================================================================================
================================================================================================

2023-07-30 15:25:49,164 - 87842 - INFO - Cleaning operation
operation_label: Cleaning with 'clean_cddb.clean_df_year' procedure
cleaning operation_label: Cleaning with 'clean_cddb.clean_df_year' procedure
Number of rows affected: 5353
Columns affected: {'year'}
Examples:
|      |   ('year', 'before') |   ('year', 'after') |
|-----:|---------------------:|--------------------:|
|  824 |                 1999 |                1999 |
| 9035 |                 2003 |                2003 |
| 9425 |                 1997 |                1997 |
| 3242 |                 1998 |                1998 |
| 2841 |                 2000 |                2000 |
================================================================================================
================================================================================================

2023-07-30 15:25:49,204 - 87842 - INFO - Cleaning operation
operation_label: Cleaning with 'clean_cddb.clean_df_title' procedure
cleaning operation_label: Cleaning with 'clean_cddb.clean_df_title' procedure
Number of rows affected: 8
Columns affected: {'title'}
Examples:
|      | ('title', 'before')   | ('title', 'after')   |
|-----:|:----------------------|:---------------------|
| 6093 |                       | N/A                  |
| 1255 |                       | N/A                  |
|  634 |                       | N/A                  |
| 7705 |                       | N/A                  |
| 2662 |                       | N/A                  |
================================================================================================
================================================================================================

2023-07-30 15:25:49,248 - 87842 - INFO - Cleaning operation
operation_label: Cleaning with 'clean_cddb.clean_df_genre_coalesce_with_category' procedure
cleaning operation_label: Cleaning with 'clean_cddb.clean_df_genre_coalesce_with_category' procedure
Number of rows affected: 3306
Columns affected: {'genre'}
Examples:
|      | ('genre', 'before')   | ('genre', 'after')   |
|-----:|:----------------------|:---------------------|
| 1828 | N/A                   | misc                 |
|  179 | N/A                   | folk                 |
| 6395 | N/A                   | rock                 |
| 6331 | N/A                   | rock                 |
| 5194 | N/A                   | rock                 |
================================================================================================
================================================================================================
```

Inspecting clean up on 'artist' field with (1) standardization to "Various" and (2) fixing encoding issues

In [ ]:
```
comps_df_sample_markdown: str = (source_df
                                 .compare(clean_df_artist_transforms_only, result_names=('before', 'after'))
                                 .sample(5, random_state=0)
                                 .to_markdown()
                                 )

print("Example diffs between before-and-after")
print(comps_df_sample_markdown)
```

```
Example diffs between before-and-after
|      | ('artist', 'before')   | ('artist', 'after')   |
|-----:|:-----------------------|:----------------------|
| 3371 | Various Artists        | Various               |
| 9401 | Los CaÃ Â±os                        | Los Caños             |
|  316 | MaÃ Â±a's                          | Maña's                |
| 6546 | various                | Various               |
| 4434 | Various Artists        | Various               |
```

In [ ]:
```
# apply schema to clean_df
try:
```

```
    validated_df = clean_cddb.schema(clean_df, lazy=True)
    logging.info("Validation success. No failure cases detected.")
except pa.errors.SchemaErrors as err:
    logging.info("Validation failure. Failure cases detected.")
    logging.debug(err)
    after_cleaning_failure_cases_df = err.failure_cases

after_cleaning_failure_cases_df = after_cleaning_failure_cases_df.pipe(
    get_check_func_descriptions, clean_cddb.schema
)

after_cleaning_failure_cases_summary = (
    after_cleaning_failure_cases_df.groupby(["column", "check"], as_index=False)
    .size()
    .sort_values(by=["column", "check"])
    .rename(columns={"size": "counts"})
)
```

```
2023-07-30 15:25:49,351 - 87842 - INFO - Validation failure. Failure cases detected.
```

Evaluation

Counts of Failure Cases Before vs After Data Cleaning

In [ ]:
```
(
failure_cases_summary
.merge(after_cleaning_failure_cases_summary,
       on=['column', 'check'],
       how='outer',
       suffixes=['_before_cleaning', '_after_cleaning']
       )
.fillna('')
)
```

Out[ ]:

	column	check	counts_before_cleaning	counts_after_cleaning
0	artist	Check for *possibly* invalid symbols.	639	
1	artist	Check for invalid artist values.	697	
2	artist	not_nullable	1	
3	category	Check for invalid categories.	89	
4	genre	Check for invalid genres.	1	
5	genre	not_nullable	3388	
6	id	Check that the length of 'id' is 6 characters.	477	
7	title	Check for *possibly* invalid symbols.	747	497.0
8	title	not_nullable	8	
9	year	Check if year is numeric.	28	
10	year	Check that year is between 1950 and 2030.	56	
11	year	dtype('Int32')	1	

Comparing `source_df` and `clean_df`

- We will actually use an intermediate dataframe `clean_df_before_drops` that has the same dimensions as our original dataframe.
  - Prior to dropping dirty records, `clean_df_before_drops` has values over-written with a prefix "REJECT_RECORD".
  - This enables easier side by side comparison.
- The final output `clean_df` will actually omit records that we intend to drop.

In [ ]:
```
source_df.head()
```

Out[ ]:

	artist	category	genre	title	tracks	year	id	merged_values
0	Backstreet Boys	blues	Pop	Millennium	Larger Than Life | I Want It That Way | Show Me The Meaning Of Being Lonely | It's Gotta Be You | I Need You Tonight | Don't Want You Back | Don't Wanna Lose You Now | The One | Back To Your Heart | Spanish Eyes | No One Else Comes Close | The Perfect Fan | I'll Be There For You | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |	NaN	10000	NaN
1	Various	data	NaN	Frankfurt Trance Vol. 04 cd1	DJ Tom Stevens VS. Fridge - Outface 2000 (Radio Mix) | Alice Deejay - Better Off Alone (Signum Remix) | Tillmann Uhrmacher Feat. Peter Ries - Bassfly (Original Mix) | DJ 2 L 8 - Too Late | Time Square - Invisible Girl (Future Breeze Remix) | Cirillo - Across The Soundline | DJ Leon & Jam X - Hold It | Sean Dexter - Synthetica (Extended Mix) | DJ BjÃ Â¶rn - On A Mission (Original Mix) | 8Voice - Music Hypnotizes 2000 | Alex Apollo - Jahr 2000 | Headroom - Utopia (Radio Mix) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |	NaN	100001	NaN
2	NO RETURN	data	Data	Self Mutilation	Do or Die | Truth and Reality | Lost | Soul Extractor | Sadistic Desire | The True Way | Fanatic Mind | Individualistic Ideal | One Life | Trail of Blood | Sect | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |	NaN	100002	NaN
3	Ã Â¤Ã Â¸Ã ÂÃ Â¦? Ã¢â ¬Ë Ã Â©Ã¢â ¬ÂºÃ¢â ¬Â¦Ã Â¤Ã Â¿Ã Â	data	Pop	Ã Â¦Ã â Ã Â³Ã Â£? Ã¢â ¬Å¾Ã Â¥Ã¢â ¬Â¡Ã ÂºÃ Â£? Ã Â®Ã Â£?Ã¢â ¬Â¹Ã Â£? Ã¢â ¬Ë Ã Â£Ã¢â ¬Å¡Ã¢â ¬Â°	Ã Â§Ã¢â ¬ÂºÃ¢â ¬Â Ã Â¥Ã Â¸Ã ÂºÃ Â£Ã¢â ¬Å¡Ã Â | Ã Â£? Ã¢â ¬Å¾Ã Â£?Ã Â¤Ã Â£?Ã¢â ¬Â¹Ã Â¨Ã Â¡Ã¢â ¬â Ã Â£? Ã Â§Ã Â¤Ã Â¼Ã Â¡Ã Â£?Ã Â£Ã Â£?Ã Â¸Ã Â£? Ã ÂªÃ Â£Ã¢â ¬Å¡Ã¢â ¬Â° | Ã Â©Ã Â¢Ã Â¨Ã Â£?Ã Â®Ã Â£? Ã ÂªÃ Â£?Ã¢â ¬Å¾Ã Â¦Ã¢â ¬â Ã Â¥ | Ã Â£?Ã Â¸Ã Â£? Ã Â Ã Â£?Ã Â Ã Â¥Ã¢â ¬Âº?Ã Â£?Ã â Ã Â£?Ã¢â ¬Å¾Ã Â£? Ã¢â ¬Å¾ | Ã Â£?Ã ÂµÃ Â£Ã¢â ¬Å¡Ã â Ã Â£?Ã¢â ¬Å¡Ã Â£? Ã¢â ¬Å¾ | Ã Â£?Ã¢â ¬Å¡Ã Â£Ã¢â ¬Å¡?Ã Â©? Ã¢â ¬â ¢Ã Â¦Ã Å Ã Â¥ | Ã Â¤Ã Â¿Ã ÂºÃ Â£?Ã Â¸Ã Â£? Ã Â¡Ã Â£?Ã Â®Ã Â¦Ã¢â ¬â Ã¢â ¬Â¦ | Ã Â§Ã¢â Â¢Ã Â½Ã Â£? Ã¢â ¬Å¾Ã Â¥Ã Â¯Ã Â«Ã Â§Ã â Ã Â¾Ã Â©Ã Â¤Ã Â¨ | Ã Â£? Ã¢â ¬Â¢Ã Â£?Ã¢â Â¢Ã Â£Ã¢â ¬Å¡Ã¢â ¬ÂºÃ Â£? Ã¢â ¬Å¾Ã Â¦Ã¢â Â¢Ã¢â ¬Å¡Ã Â¤Ã Â»Ã Â£ | Ã Â¥Ã Â¤Ã â Ã Â¨Ã Â¡Ã â Ã Â¥Ã â Ã¢â ¬â Ã Â¨Ã Â»Ã Â | Ã Â£? Ã¢â ¬Å¡Ã Â£?Ã ÂªÃ Â£?Ã Â¸Ã Â£Ã¢â ¬Å¡Ã¢â ¬â ¢Ã Â¦Ã¢â ¬Å¾Ã¢â ¬ÂºÃ Â£? Ã¢â Â¢Ã Â£Ã¢â ¬Å¡Ã¢â ¬Â¹Ã Â§Ã Â§? | Ã Â©? Ã¢â ¬â ¢Ã Â¦Ã Å Ã Â¥Ã Â¨Ã Â²Ã Â´Ã Â¦Ã¢â ¬â ? | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |	1989	100003	NaN
4	Emanuel	data	Data	Felicidade	Felicidade quando o telefone toca | Vem bailar o tic tic | Quero que sejas minha e de mais ninguem | Eu sei que me amas | O melhor que hÃ Â¡ | Minha vizinha deixa me a bater mal | S. JoÃ Â£o Ã Â© foliÃ Â£o | SÃ Â³ quero o teu carinho | tudo farei para ter a tua paixÃ Â£o | serÃ Â¡s sempre minha | Vem bailar o tic tic verÃ Â§Ã Â£o dance | Mix | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |	1998	100004	NaN

In [ ]:

```python
clean_df_before_drops.head()
```

Out[ ]:

	artist	category	genre	title	tracks	year	id	merged_values
0	Backstreet Boys	blues	Pop	Millennium	Larger Than Life | I Want It That Way | Show Me The Meaning Of Being Lonely | It's Gotta Be You | I Need You Tonight | Don't Want You Back | Don't Wanna Lose You Now | The One | Back To Your Heart | Spanish Eyes | No One Else Comes Close | The Perfect Fan | I'll Be There For You | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |	<NA>	010000	NaN
1	Various	N/A	N/A	Frankfurt Trance Vol. 04 cd1	DJ Tom Stevens VS. Fridge - Outface 2000 (Radio Mix) | Alice Deejay - Better Off Alone (Signum Remix) | Tillmann Uhrmacher Feat. Peter Ries - Bassfly (Original Mix) | DJ 2 L 8 - Too Late | Time Square - Invisible Girl (Future Breeze Remix) | Cirillo - Across The Soundline | DJ Leon & Jam X - Hold It | Sean Dexter - Synthetica (Extended Mix) | DJ BjÃ Â¶rn - On A Mission (Original Mix) | 8Voice - Music Hypnotizes 2000 | Alex Apollo - Jahr 2000 | Headroom - Utopia (Radio Mix) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |	<NA>	100001	NaN
2	NO RETURN	N/A	N/A	Self Mutilation	Do or Die | Truth and Reality | Lost | Soul Extractor | Sadistic Desire | The True Way | Fanatic Mind | Individualistic Ideal | One Life | Trail of Blood | Sect | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |	<NA>	100002	NaN
3	REJECT_ROW - invalid artist	N/A	REJECT_ROW - invalid artist	REJECT_ROW - invalid artist	REJECT_ROW - invalid artist	<NA>	REJECT_ROW - invalid artist	REJECT_ROW - invalid artist
4	Emanuel	N/A	N/A	Felicidade	Felicidade quando o telefone toca | Vem bailar o tic tic | Quero que sejas minha e de mais ninguem | Eu sei que me amas | O melhor que hÃ Â¡ | Minha vizinha deixa me a bater mal | S. JoÃ Â£o Ã Â© foliÃ Â£o | SÃ Â³ quero o teu carinho | tudo farei para ter a tua paixÃ Â£o | serÃ Â¡s sempre minha | Vem bailar o tic tic verÃ Â§Ã Â£o dance | Mix | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |	1998	100004	NaN

In [ ]:

```python
pd.set_option('display.max_colwidth', 50)

comps_df = (
source_df.sort_index()
 .compare(clean_df_before_drops.sort_index(), result_names=('before_cleaning', 'after_cleaning'))
 .astype('object')
 .fillna('')
 )
```

Show before/after comps

In [ ]:

```python
pd.set_option('display.max_colwidth', None)

columns_to_compare = ['artist', 'category', 'genre', 'title', 'tracks', 'year', 'id']

comps_df_formatted = (
comps_df
 .astype(str)
 .stack()
 .reset_index()
 .rename(columns={'level_0': 'row_id', 'level_1': 'before_or_after'})
 .drop(columns=['merged_values'])
 .groupby(['row_id'], as_index=False)
    [columns_to_compare]
    .agg(lambda row: '  =>  '.join(row))
 .replace('^(  =>  )$', '', regex=True)
)

comps_df_formatted.sample(5, random_state=1)
```

Out[ ]:

	row_id	artist	category	genre	title	tracks	year	id
4320	4904						1994 => 1994	
6257	7096			=> rock				
1924	2187						1977 => 1977	
6332	7184	Die schÃ Â¶nsten Westernmelodien => Die schönsten Westernmelodien						
3350	3813						2000 => 2000	

In [ ]:
```
print("Number of rows changed per column")

for column in clean_df.columns:
    n_rows_changed = comps_df[column][comps_df[column]['before_cleaning'] != comps_df[column]['after_cleaning']].shape[0]
    print(f"{column:.<15}{n_rows_changed}")
```

```
Number of rows changed per column
artist.........1326
category.......421
genre..........3684
title..........355
tracks.........347
year...........5231
id.............808
```

Sample transformations

- Here we can see that we transform "Various Artists" and "" to "Various".
- We also fixed invalid characters converting text from "JÃ¶rg Hilbert & Felix Janosa" to "Jörg Hilbert & Felix Janosa".
- Later, we will do a more comprehensive before/after analysis after applying all the cleaning transformations

In [ ]:
```
example_idxs = [7629, 1822, 117, 4129]
source_df.compare(clean_df_artist_transforms_only).loc[example_idxs, :].fillna(pd.NA)
```

Out[ ]:

	artist	
	self	**other**
7629	Various Artists	Various
1822	JÃ Â¶rg Hilbert & Felix Janosa	Jörg Hilbert & Felix Janosa
117	Various Artists	Various
4129	<various>	Various

In [ ]:
```
comps_df.sample(10, random_state=3)
```

Out[ ]:

	artist		category		genre		title	
	before_cleaning	**after_cleaning**	**before_cleaning**	**after_cleaning**	**before_cleaning**	**after_cleaning**	**before_cleaning**	**after_cleaning**
6337						rock		
7462								
906								
6771								
3271						misc		
7271								
8111								
9438	Ã Å Ã Â³Ã Â°Ã Â§Ã Â¨Ã Â«Ã Â³Ã Â¨ International	REJECT_ROW - invalid artist	misc	N/A	Russian Pop	REJECT_ROW - invalid artist	Ã Å Ã Â³Ã Â°Ã Â§Ã Â¨Ã Â«Ã Â³Ã Â¨ +	REJECT_ROW - invalid artist
3909						misc		
8211								

In [ ]:
```
comps_df.sample(5, random_state=4)
```

Out[ ]:

	artist		category		genre		title		tracks			
	before_cleaning	**after_cleaning**	**before_cleaning**	**after_cleaning**	**before_cleaning**	**after_cleaning**	**before_cleaning**	**after_cleaning**	**before_cleaning**	**after_cleaning**	**before_cleaning**	**after**
6112											1999	
6082	Various Artists	Various									1991	
4633											1999	
1034						jazz						
1581	Various Artists	Various									1992	

Smaller before/after example

- Here we can see that we transform "Various Artists" and "" to "Various".
- We also fixed invalid characters converting text from "JÃ¶rg Hilbert & Felix Janosa" to "Jörg Hilbert & Felix Janosa".
- Later, we will do a more comprehensive before/after analysis after applying all the cleaning transformations

In [ ]:
```
example_idxs = [7629, 1822, 117, 4129]
source_df.compare(clean_df_artist_transforms_only).loc[example_idxs, :].fillna(pd.NA)
```

Out[ ]:

	artist	
	self	other
7629	Various Artists	Various
1822	JÃ Â¶rg Hilbert & Felix Janosa	Jörg Hilbert & Felix Janosa
117	Various Artists	Various
4129	<various>	Various

Sample transformations

- Here we can see that we transform "Various Artists" and "" to "Various".
- We also fixed invalid characters converting text from "JÃ¶rg Hilbert & Felix Janosa" to "Jörg Hilbert & Felix Janosa".
- Later, we will do a more comprehensive before/after analysis after applying all the cleaning transformations

In [ ]:
```
example_idxs = [7629, 1822, 117, 4129]
source_df.compare(clean_df_artist_transforms_only).loc[example_idxs, :].fillna(pd.NA)
```

Out[ ]:

	artist	
	self	other
7629	Various Artists	Various
1822	JÃ Â¶rg Hilbert & Felix Janosa	Jörg Hilbert & Felix Janosa
117	Various Artists	Various
4129	<various>	Various

Transform to track-level data

We want a separate track-level dataset that can be joined to the album-level data in `clean_df`.

In [ ]:
```
# Transform to track-level data
track_level_df = (
    # Start with original dataframe
    clean_df
    # Split 'tracks' on pipe into an array; we can "explode" it later
    .pipe(lambda _df: _df.assign(tracks=_df["tracks"].str.split("|")))
    # "explode"/expand from "tracks" array in to 1 observation per track
    # perform a self-join to CD data set; the CD-level data will repeat for each track
    .pipe(
        lambda _df: _df.merge(
            _df["tracks"].explode(), left_index=True, right_index=True
        )
    )
    .pipe(df_to_var, "df_after_explode")
    # Make new 'tracks' field; strip ' ' empty space track names to '' empty string
    .pipe(lambda _df: _df.assign(tracks=_df["tracks_y"].str.strip()))
    # Don't need these fields anymore
    .drop(columns=["tracks_x", "tracks_y"])
    # Filter out empty string track names
    .query("tracks!=''")
    .pipe(df_to_var, "df_after_empty_track_name_filter")
    .reset_index(drop=True)
    .loc[:, ['id', 'tracks']]
    .reset_index()
    .rename(columns={'id': 'album_row_id',
                     'index': 'track_id',
                     'tracks': 'track_name',
                     }
            )
)
```

In [ ]:
```
# Demo joining track_level_df and clean_df
(track_level_df
 .head(15)
 .merge(clean_df, left_on=['album_row_id'], right_on=['id'], how='inner')
 .loc[:, ['album_row_id', 'track_id', 'title', 'track_name']]
)
```

Out[ ]:

	album_row_id	track_id	title	track_name
0	010000	0	Millennium	Larger Than Life
1	010000	1	Millennium	I Want It That Way
2	010000	2	Millennium	Show Me The Meaning Of Being Lonely
3	010000	3	Millennium	It's Gotta Be You
4	010000	4	Millennium	I Need You Tonight
5	010000	5	Millennium	Don't Want You Back
6	010000	6	Millennium	Don't Wanna Lose You Now
7	010000	7	Millennium	The One
8	010000	8	Millennium	Back To Your Heart
9	010000	9	Millennium	Spanish Eyes
10	010000	10	Millennium	No One Else Comes Close
11	010000	11	Millennium	The Perfect Fan
12	010000	12	Millennium	I'll Be There For You
13	100001	13	Frankfurt Trance Vol. 04 cd1	DJ Tom Stevens VS. Fridge - Outface 2000 (Radio Mix)
14	100001	14	Frankfurt Trance Vol. 04 cd1	Alice Deejay - Better Off Alone (Signum Remix)

End