

Enriching a Dataset - Inner Join Implementation

The screenshot shows a web browser window with the URL `campus.datacamp.com/courses/joining-data-with-pandas/merging-tables-with-different-join-types/ex=3`. The page is titled "Enriching a dataset" and contains the following instructions:

Setting `how='left'` with the `.merge()` method is a useful technique for enriching or enhancing a dataset with additional information from a different table. In this exercise, you will start off with a sample of movie data from the movie series *Toy Story*. Your goal is to enrich this data by adding the marketing tag line for each movie. You will compare the results of a left join versus an inner join.

The `toy_story` DataFrame contains the *Toy Story* movies. The `toy_story` and `taglines` DataFrames have been loaded for you.

Instructions 2/2

- Merge `toy_story` and `taglines` on the `id` column with a left join, and save the result as `toystory_tag`.
- With `toy_story` as the left table, merge to it `taglines` on the `id` column with an inner join, and save as `toystory_tag`.

The code editor shows the following Python code:

```
1 # Merge the toy_story and taglines tables with a inner join
2 toystory_tag = ----
3
4 # Print the rows and shape of toystory_tag
5 print(toystory_tag)
6 print(toystory_tag.shape)
```

The Python Shell output shows the following DataFrame:

	id	title	popularity	release_date	tagline
0	10193	Toy Story 2	97.995	2010-06-10	No toy gets left behind.
1	863	Toy Story 2	75.525	1999-10-30	The Toys are back!
2	862	Toy Story	73.640	1995-10-30	NaN

Question:

In this exercise, you are tasked with enriching a dataset of movies from the Toy Story series by adding taglines using the `merge` method. You will perform an inner join and analyze the resulting DataFrame.

Instructions:

- Merge the `toy_story` and `taglines` tables using an inner join on the `id` column.
- Save the resulting DataFrame to `toystory_tag`.
- Print the contents and the shape of the resulting DataFrame.

Answer:

```
# Step 1: Merge `toy_story` and `taglines` with an inner join
toystory_tag = toy_story.merge(taglines, on='id', how='inner')
```

```
# Step 2: Print the rows and shape of the merged DataFrame
print(toystory_tag)
print(toystory_tag.shape)
```

Explanation of the Code:

- `toy_story.merge(taglines, on='id', how='inner')`: This performs an inner join between the `toy_story` and `taglines` tables on the `id` column. It

ensures that only rows with matching `id` values in both tables are included in the result.

2. ``print(toystory_tag)``: Displays the contents of the merged DataFrame to verify the inner join operation.

3. ``print(toystory_tag.shape)``: Outputs the shape of the resulting DataFrame (number of rows and columns), which can be used to verify that only matching rows are included.