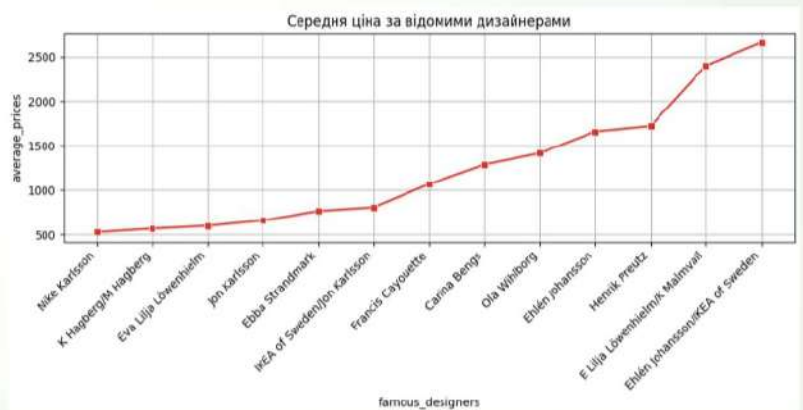


```

## Побудова кривої Середня ціна за відомими дизайнерами
average_prices_by_Fdesigner = famous_designers_data.groupby('designer_clean')['price'].mean().sort_values()
plt.figure(figsize=(10, 5))
sns.lineplot(x=average_prices_by_Fdesigner.index, y=average_prices_by_Fdesigner.values,
             marker='s', color='red')
plt.title('Середня ціна за відомими дизайнерами')
plt.xlabel('famous_designers')
plt.ylabel('average_prices')
plt.xticks(rotation=45, ha='right')
plt.grid(True)
plt.tight_layout()
plt.show()

```



```

## Побудова кривої Середня ціна за меншвідомими дизайнерами
average_prices_by_Ldesigner = less_known_designers_data.groupby('designer_clean')['price'].mean().sort_values()
plt.figure(figsize=(10, 5))
sns.lineplot(x=average_prices_by_Ldesigner.index, y=average_prices_by_Ldesigner.values, marker='s', color='green')
plt.title('Середня ціна за меншвідомими дизайнерами')
plt.xlabel('less_known_designers')
plt.ylabel('average_prices')
plt.xticks(rotation=60, ha='right', fontsize=8)
plt.grid(True)
plt.tight_layout()
plt.show()

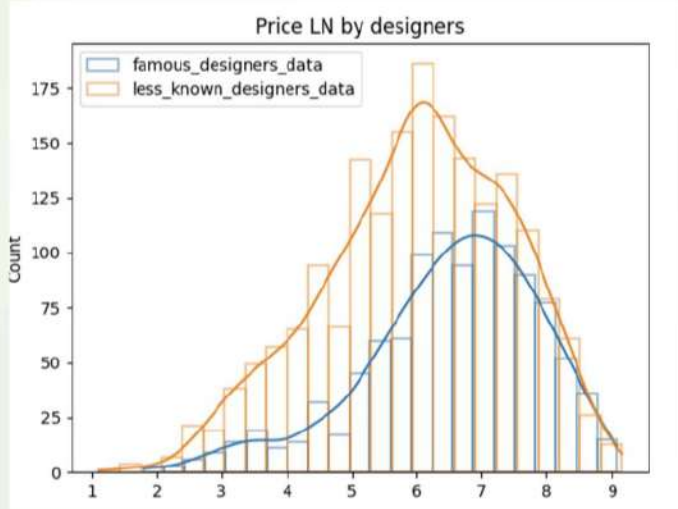
```



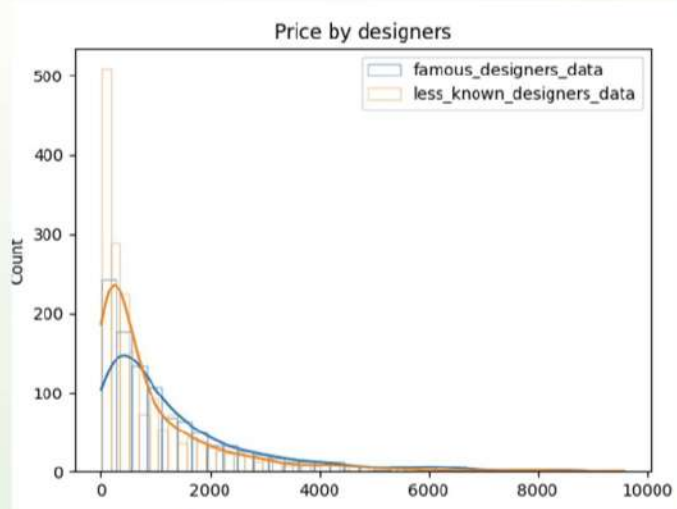
Графіки накладаються по кількостях і відповідно ми не можемо сказати, що ціна не дуже відрізняється по дизайнерах.

```
### звичайна ціна розподіл за дизайнерами
show_2_designers_histplot(famous_prices_values, less_known_prices_values,
                           plotTitle: 'Price by designers',
                           xlabel: 'famous_designers_data',
                           ylabel: 'less_known_designers_data')

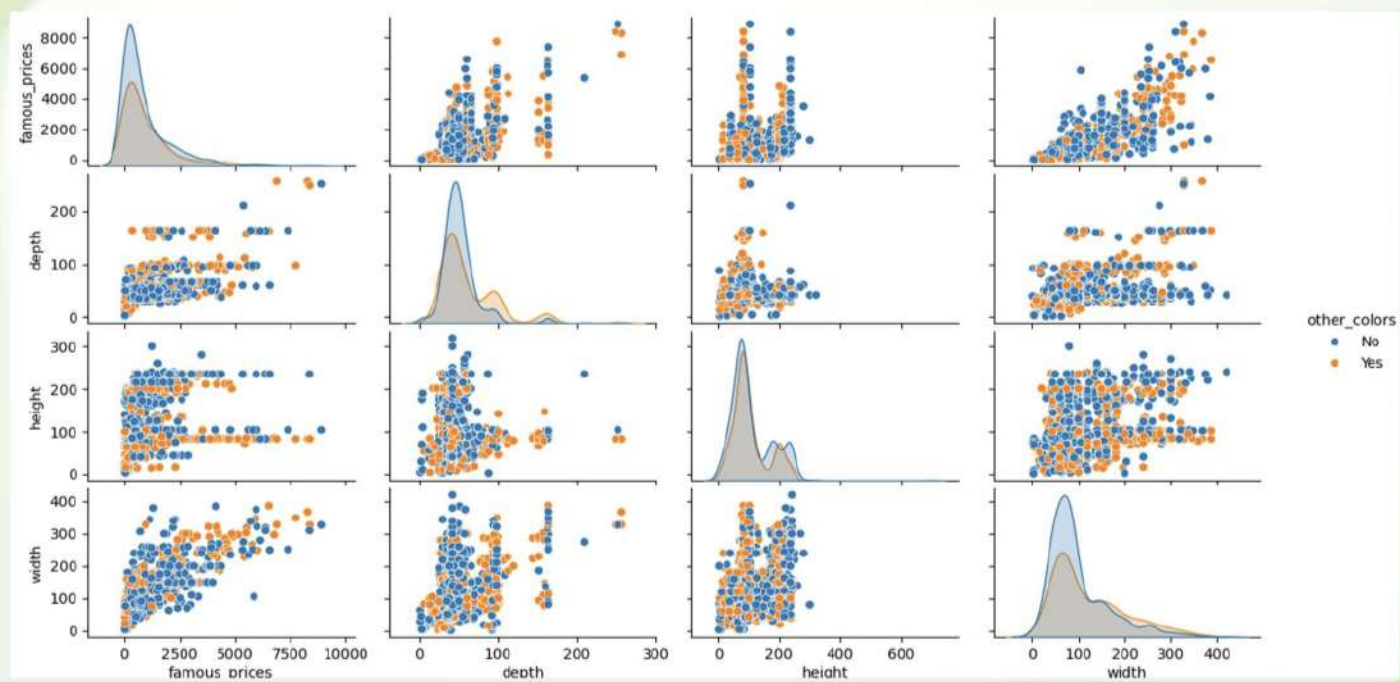
famous_prices_values_LN = famous_designers_data['price_ln'].values
less_known_prices_values_LN = less_known_designers_data['price_ln'].values
```

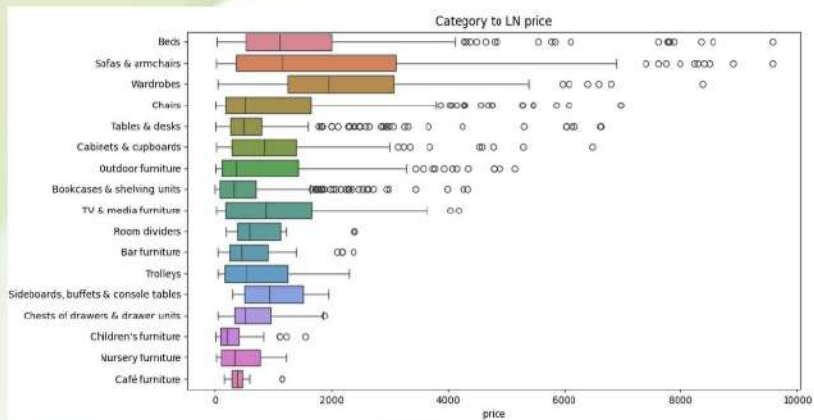


```
### прологарифмована ціна розподіл за дизайнерами
show_2_designers_histplot(famous_prices_values_LN, less_known_prices_values_LN,
                           plotTitle: 'Price LN by designers',
                           xlabel: 'famous_designers_data',
                           ylabel: 'less_known_designers_data')
```



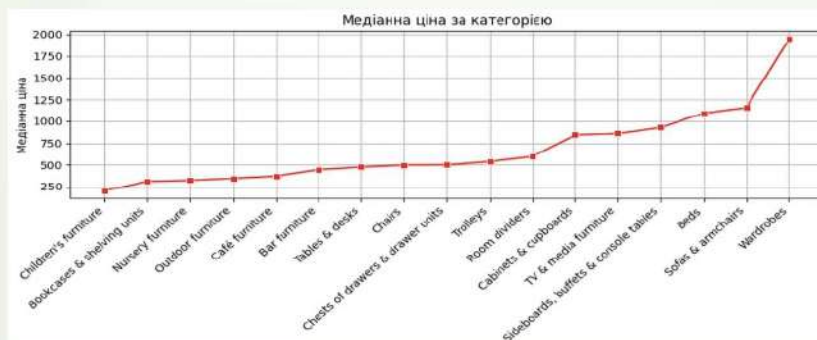
Візуалізуємо залежність між ціною та параметрами (глибина, висота, ширина)





```
plt.subplots(figsize=(10, 5))
category_count = df['category'].value_counts().index
sns.boxplot(data=df.sort_values(by='price', ascending=False), x='price', y='category', hue='category')
plt.title('Category to LN price')
plt.show()
```

Найбільше викидів від середньої ціни ми бачимо у категоріях Sofas & armchairs та Beds

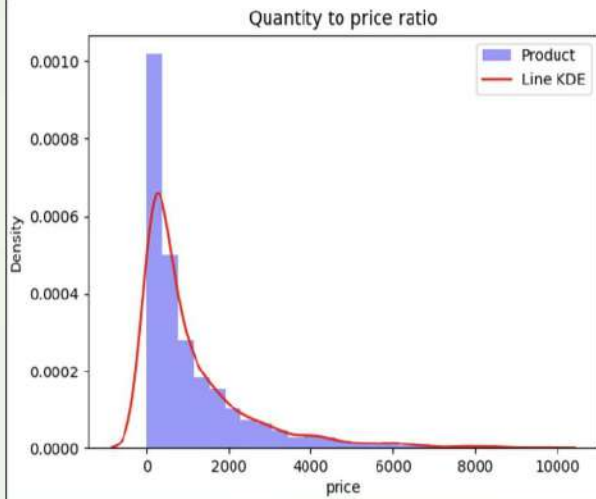


```
# Обчислення середніх цін по категоріям
median_prices_by_category = df.groupby('category')['price'].median().sort_values()

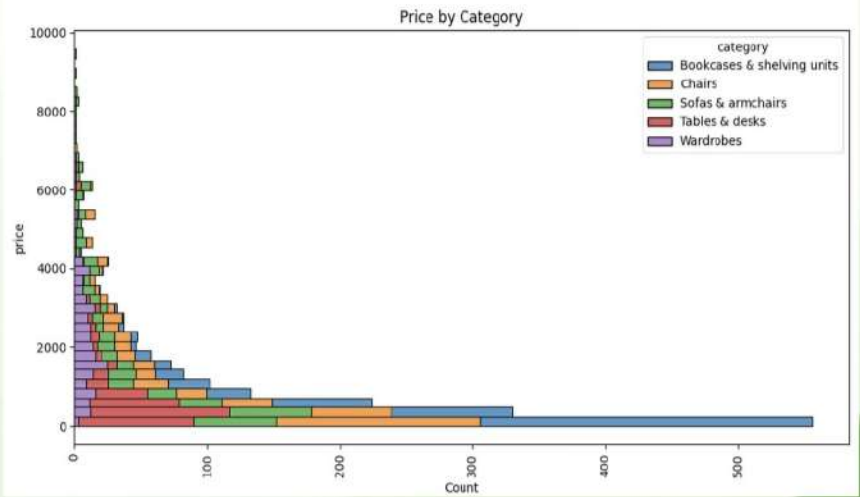
# Нормування графіка
plt.figure(figsize=(10, 5))
sns.lineplot(x=median_prices_by_category.index, y=median_prices_by_category.values, marker='s', color='red')
plt.title('Медіанна ціна за категорією')
plt.xlabel('Категорія')
plt.ylabel('Медіанна ціна')
plt.xticks(rotation=45, loc='right')
plt.grid(True)
plt.tight_layout()
plt.show()
```

Найнижча медіанна ціна за категорією Children's furniture, а найвижча - Wardrobes

Підтверджуємо дані. Розподіл кількості товарів в категорії в залежності від ціни показує, що основна кількість товарів магазину знаходиться у низькому ціновому сегменті

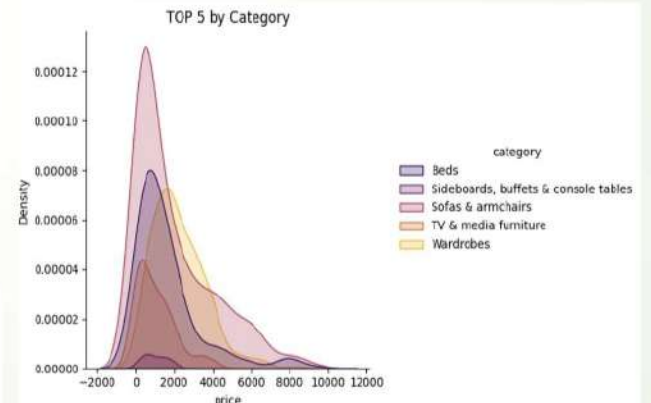
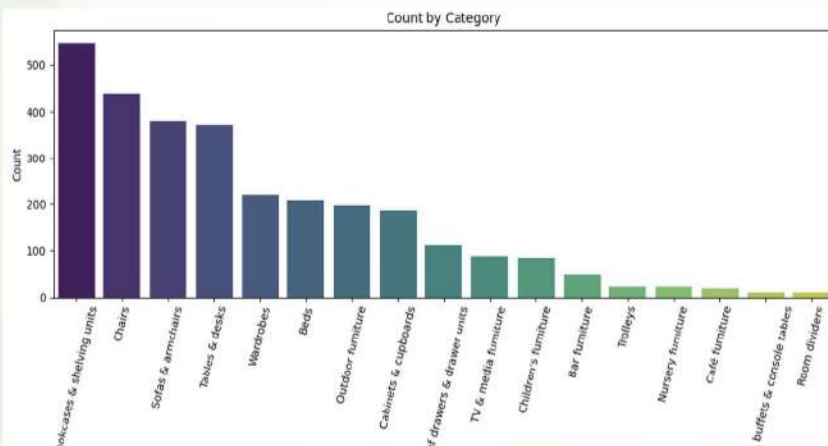


```
bins=40, element='bars', legend=True, color='orange')
plt.xticks(rotation=90)
plt.title('Price by Category')
plt.show()
```



```
# # Дивимось найдорожчі категорії (TOP 5)
median_prices = df.groupby('category')['price'].median()
top_5_categories = median_prices.sort_values(ascending=False).head(5).index.tolist()
top_5_data = df[df['category'].isin(top_5_categories)]

sns.displot(x='price', hue='category', data=top_5_data, palette='inferno', fill=True, kind='kde')
plt.title('\nTOP 5 by Category')
plt.show()
```



```
# # Додатково переглядаємо : Яких найменувань у каталозі найбільше?
plt.figure(figsize=(12, 6))
category_order = df['category'].value_counts().index
sns.countplot(x='category', data=df, order=category_order, palette='viridis')
plt.xlabel('category')
plt.ylabel('Count')
plt.xticks(rotation=75)
plt.title('Count by Category')
plt.show()
```