

## PIZZA SHOP SALES ANALYSIS PYTHON (PDF)

### KPI's

```
total_revenue = df["total_price"].sum()
total_pizzas_sold = df["quantity"].sum()
total_orders = df["order_id"].nunique()
avg_order_value = total_revenue / total_orders
avg_pizza_per_order = total_pizzas_sold / total_orders

print(f"Total Revenue: ${total_revenue:,.2f}")
print(f"Total Pizzas Sold: {total_pizzas_sold:,}")
print(f"Total Orders: {total_orders:,}")
print(f"Avg Order Value: ${avg_order_value:,.2f}")
print(f"Avg Pizza Per Order: {avg_pizza_per_order:,.2f}")
```

Total Revenue: \$817,860.05  
Total Pizzas Sold: 49,574.0  
Total Orders: 21,350  
Avg Order Value: \$38.31  
Avg Pizza Per Order: 2.32

### INGRDIENT ANALYSIS

```
ingredients = (
    df["pizza_ingredients"]
    .str.split(",")
    .explode()
    .str.strip()
    .value_counts()
    .reset_index()
    .rename(columns = {"index": "count", "pizza_ingredients": "ingredients"})
)
print(ingredients.head(10))
```

	ingredients	count
0	Garlic	27422
1	Tomatoes	26601
2	Red Onions	19547
3	Red Peppers	16284
4	Mozzarella Cheese	10333
5	Pepperoni	10300
6	Spinach	10012
7	Mushrooms	9624
8	Chicken	8443
9	Capocollo	6572

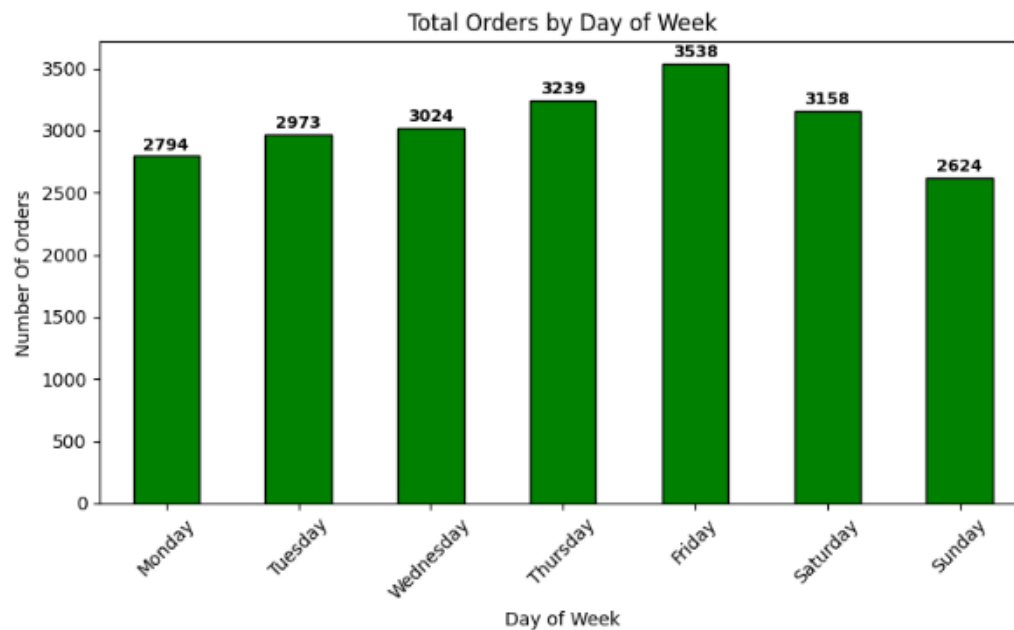
## DAILY TREND - TOTAL ORDERS

```
df["order_date"] = pd.to_datetime(df["order_date"], dayfirst = True)
df["day_name"] = df["order_date"].dt.day_name()
weekday_order = ["Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday", "Sunday"]
df["day_name"] = pd.Categorical(df["day_name"], categories = weekday_order, ordered=True)
orders_by_day = df.groupby("day_name", observed=False)["order_id"].nunique()
ax = orders_by_day.plot(kind="bar", figsize=(8,5), color="green", edgecolor="black")

plt.title("Total Orders by Day of Week")
plt.xlabel("Day of Week")
plt.ylabel("Number Of Orders")
plt.xticks(rotation=45)

for i, val in enumerate(orders_by_day):
    plt.text(i, val + 20, str(val), ha="center", va="bottom", fontsize=9, fontweight="bold")

plt.tight_layout()
plt.show()
```



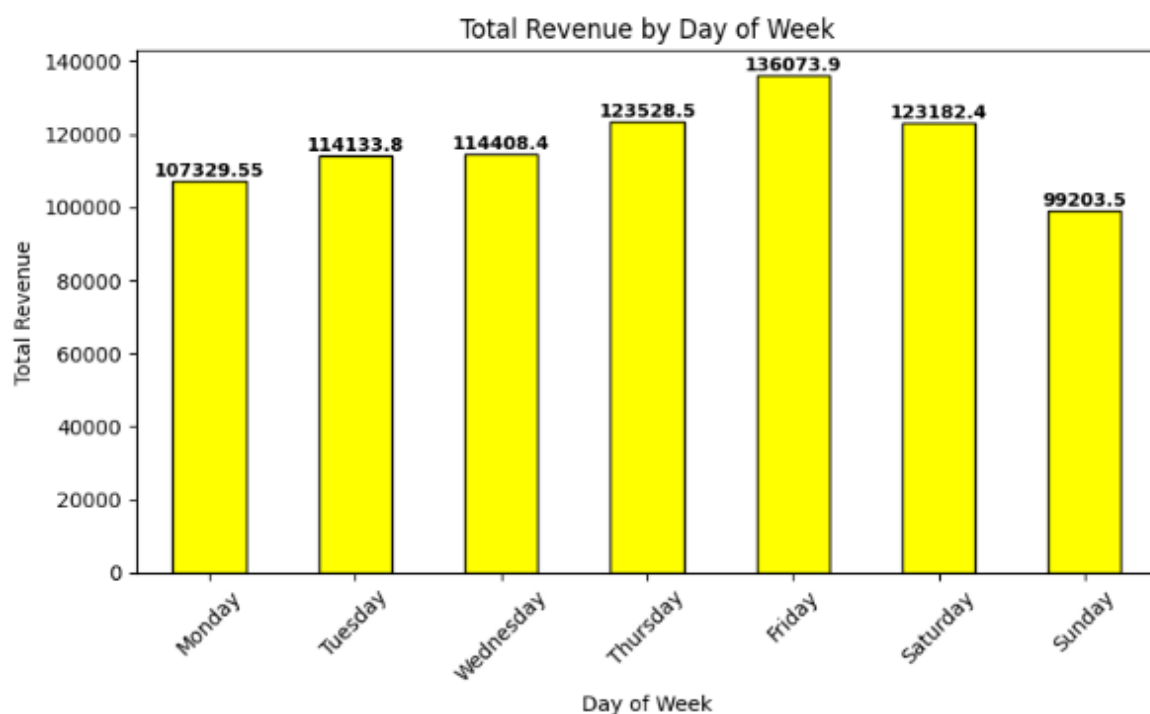
## DAILY TREND - TOTAL REVENUE

```
df["order_date"] = pd.to_datetime(df["order_date"], dayfirst=True)
df["day_name"] = df["order_date"].dt.day_name()
weekday_order = ["Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday", "Sunday"]
df["day_name"] = pd.Categorical(df["day_name"], categories=weekday_order, ordered=True)
orders_by_day = df.groupby(df["day_name"], observed=False)["total_price"].sum()
ax = orders_by_day.plot(kind="bar", figsize=(8,5), color="yellow", edgecolor="black")

plt.title("Total Revenue by Day of Week")
plt.xlabel("Day of Week")
plt.ylabel("Total Revenue")
plt.xticks(rotation=45)

for i, val in enumerate(orders_by_day):
    plt.text(i, val + 20, str(val), ha="center", va="bottom", fontsize=9, fontweight="bold")

plt.tight_layout()
plt.show()
```



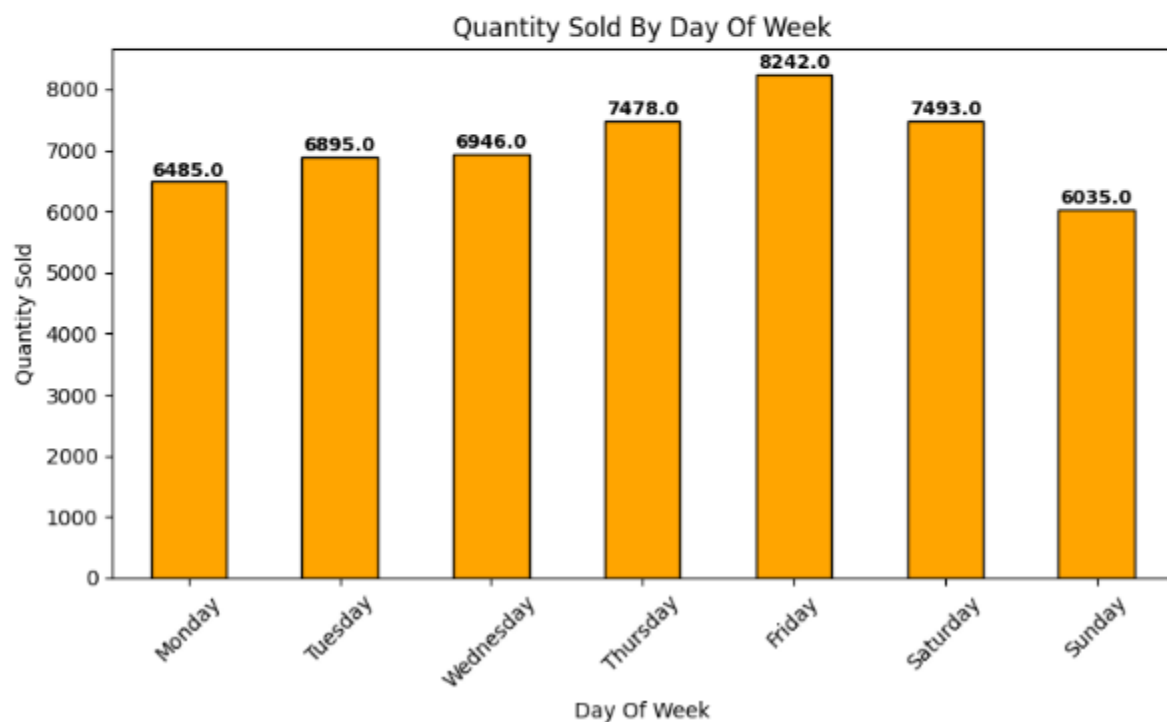
## Daily Trend - By Quantity

```
df["order_date"] = pd.to_datetime(df["order_date"], dayfirst = True)
df["day_name"] = df["order_date"].dt.day_name()
weekday_order = ["Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday", "Sunday"]
df["day_name"] = pd.Categorical(df["day_name"], categories=weekday_order, ordered=True)
quantity_sold_by_day = df.groupby("day_name", observed=False)["quantity"].sum()
ax = quantity_sold_by_day.plot(kind="bar", figsize=(8,5), color="orange", edgecolor="black")

plt.title("Quantity Sold By Day Of Week")
plt.xlabel("Day Of Week")
plt.ylabel("Quantity Sold")
plt.xticks(rotation=45)

for i, val in enumerate(quantity_sold_by_day):
    plt.text(i, val + 20, str(val), ha="center", va="bottom", fontsize=9, fontweight="bold")

plt.tight_layout()
plt.show()
```



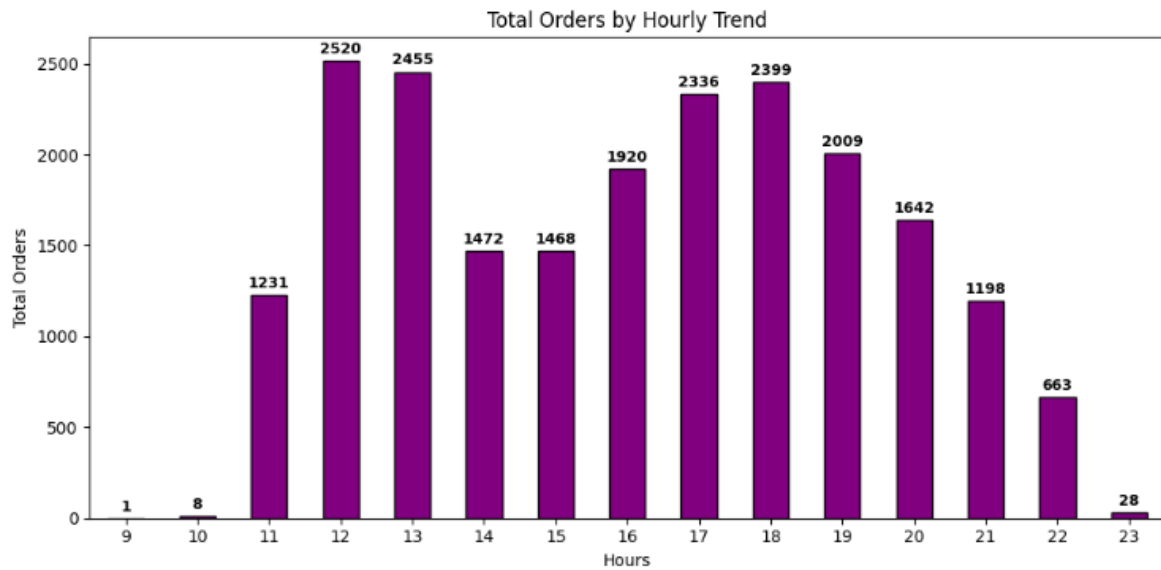
## Hourly Trend - Total Orders

```
] df["order_time"] = pd.to_datetime(df["order_time"], dayfirst=True)
df["order_hour"] = df["order_time"].dt.hour
orders_by_hour = df.groupby("order_hour", observed=False)["order_id"].nunique()
ax = orders_by_hour.plot(kind="bar", figsize=(10,5), color="purple", edgecolor="black")

plt.title("Total Orders by Hourly Trend")
plt.xlabel("Hours")
plt.ylabel("Total Orders")
plt.xticks(rotation=0)

for i, val in enumerate(orders_by_hour):
    plt.text(i, val + 20, str(val), ha="center", va="bottom", fontsize=9, fontweight="bold")

plt.tight_layout()
plt.show()
```



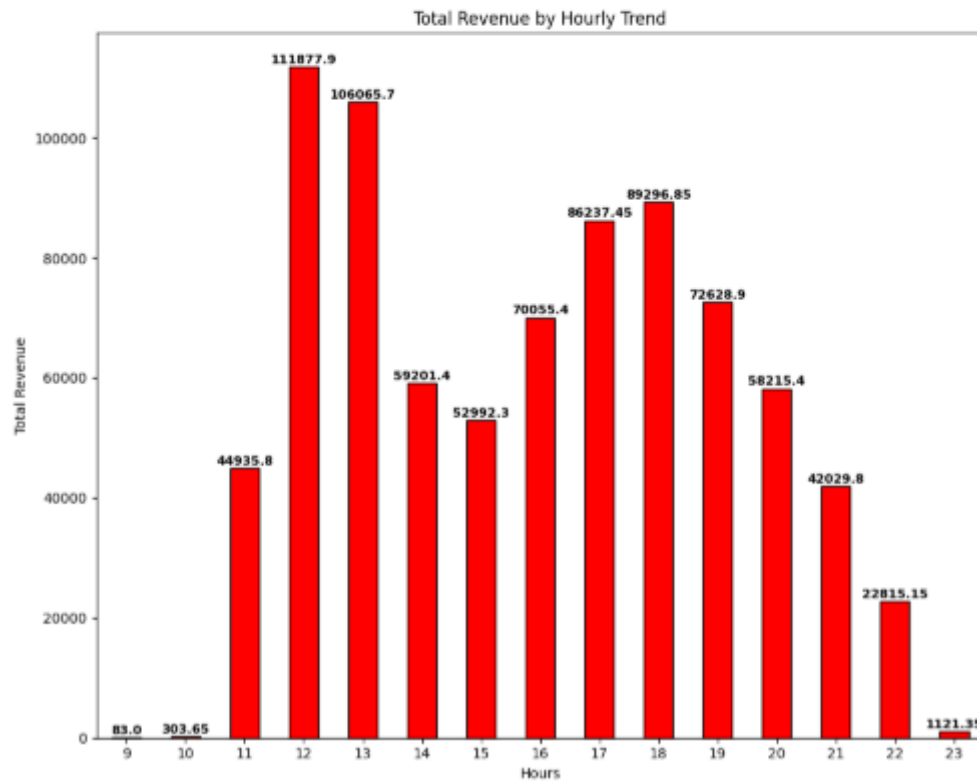
## Hourly Trend - Total Revenue

```
df["order_time"] = pd.to_datetime(df["order_time"], dayfirst=True)
df["order_hour"] = df["order_time"].dt.hour
sales_by_hour = df.groupby("order_hour", observed=False)["total_price"].sum()
ax = sales_by_hour.plot(kind="bar", figsize=(18,8), color="red", edgecolor="black")

plt.title("Total Revenue by Hourly Trend")
plt.xlabel("Hours")
plt.ylabel("Total Revenue")
plt.xticks(rotation=0)

for i, val in enumerate(sales_by_hour):
    plt.text(i, val + 30, str(val), ha="center", va="bottom", fontsize=9, fontweight="bold")

plt.tight_layout()
plt.show()
```



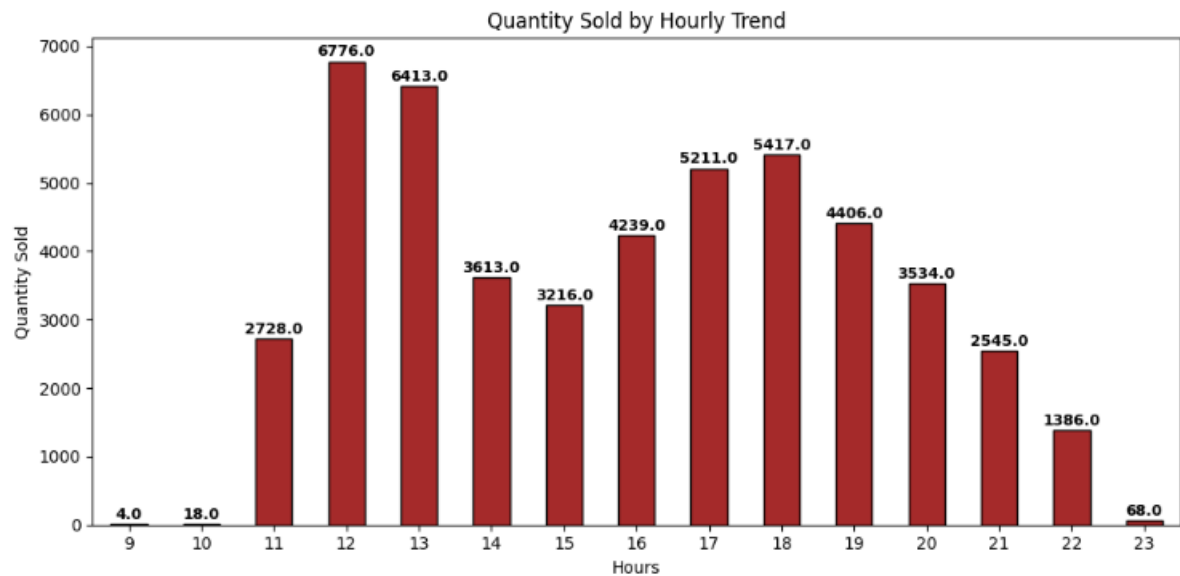
## Hourly Trend - Quantity Sold

```
df["order_time"] = pd.to_datetime(df["order_time"], dayfirst=True)
df["order_hour"] = df["order_time"].dt.hour
quantity_sold_by_hour = df.groupby("order_hour", observed=False)["quantity"].sum()
ax = quantity_sold_by_hour.plot(kind="bar", figsize=(10,5), color="brown", edgecolor="black")

plt.title("Quantity Sold by Hourly Trend")
plt.xlabel("Hours")
plt.ylabel("Quantity Sold")
plt.xticks(rotation=0)

for i, val in enumerate(quantity_sold_by_hour):
    plt.text(i, val + 20, str(val), ha="center", va="bottom", fontsize=9, fontweight="bold")

plt.tight_layout()
plt.show()
```



## Monthly Trend - Total Orders

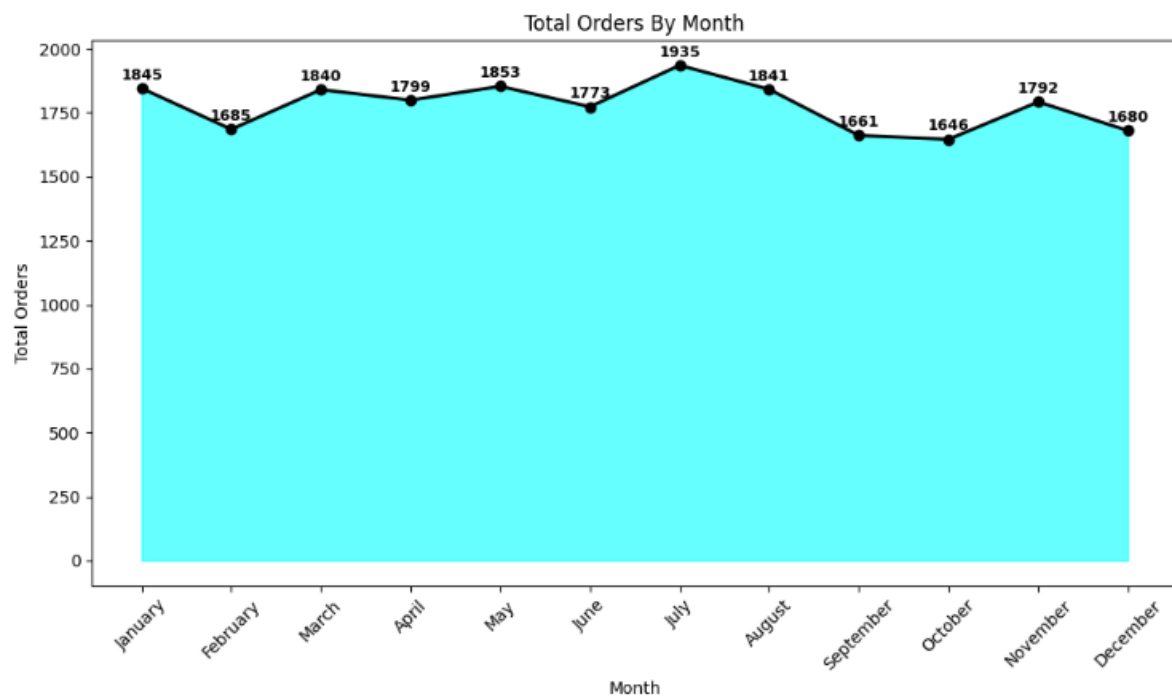
```
df["order_date"] = pd.to_datetime(df["order_date"], dayfirst=True)
df["month_name"] = df["order_date"].dt.month_name()
month_order = ["January", "February", "March", "April", "May", "June",
               "July", "August", "September", "October", "November", "December"]
df["month_name"] = pd.Categorical(df["month_name"], categories=month_order, ordered=True)
orders_by_month = df.groupby("month_name", observed=False)["order_id"].nunique()

plt.figure(figsize=(10,6))
plt.fill_between(orders_by_month.index, orders_by_month.values, color="aqua", alpha=0.6)
plt.plot(orders_by_month.index, orders_by_month.values, color="black", linewidth=2, marker="o")

plt.title("Total Orders By Month")
plt.xlabel("Month")
plt.ylabel("Total Orders")
plt.xticks(rotation=45)

for i, val in enumerate(orders_by_month):
    plt.text(i, val + 20, str(val), ha="center", va="bottom", fontsize=9, fontweight="bold")

plt.tight_layout()
plt.show()
```





## Monthly Trend - Total Revenue

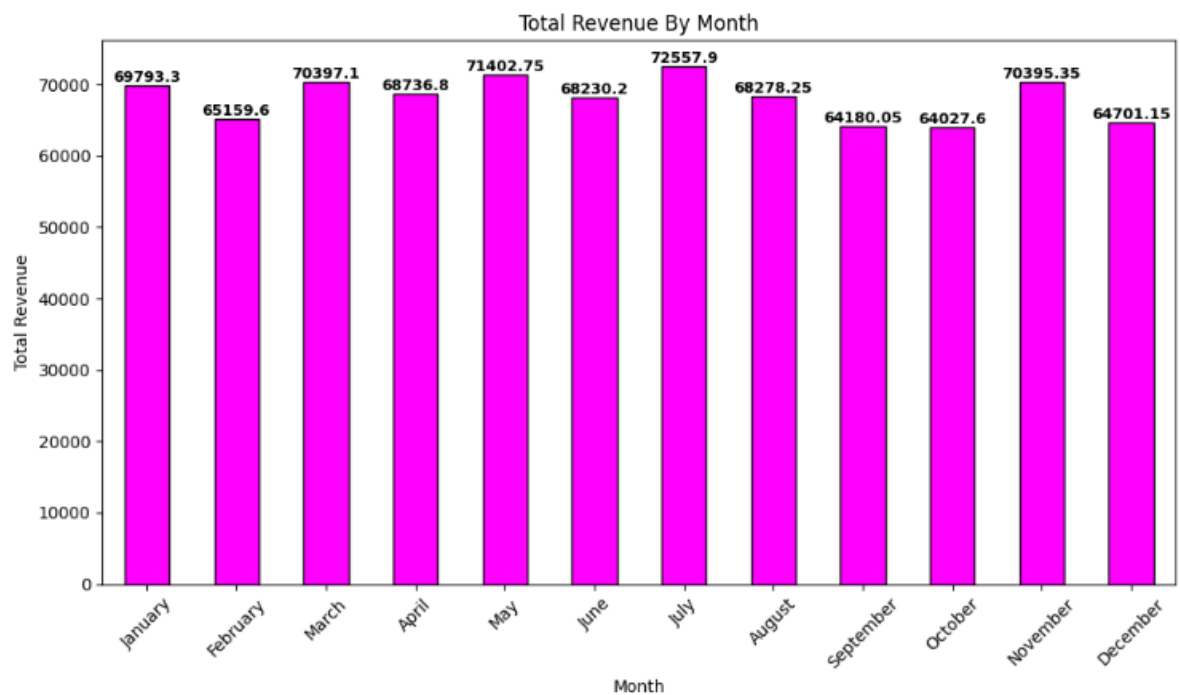
```
df["order_date"] = pd.to_datetime(df["order_date"], dayfirst=True)
df["month_name"] = df["order_date"].dt.month_name()
month_order = ["January", "February", "March", "April", "May", "June",
               "July", "August", "September", "October", "November", "December"]
df["month_name"] = pd.Categorical(df["month_name"], categories=month_order, ordered=True)
sales_by_month = df.groupby("month_name", observed=False)["total_price"].sum()

ax = sales_by_month.plot(kind="bar", figsize=(10,6), color="fuchsia", edgecolor="black")

plt.title("Total Revenue By Month")
plt.xlabel("Month")
plt.ylabel("Total Revenue")
plt.xticks(rotation=45)

for i, val in enumerate(sales_by_month):
    plt.text(i, val + 20, str(val), ha="center", va="bottom", fontsize=9, fontweight="bold")

plt.tight_layout()
plt.show()
```



## Monthly Trend - Quantity sold

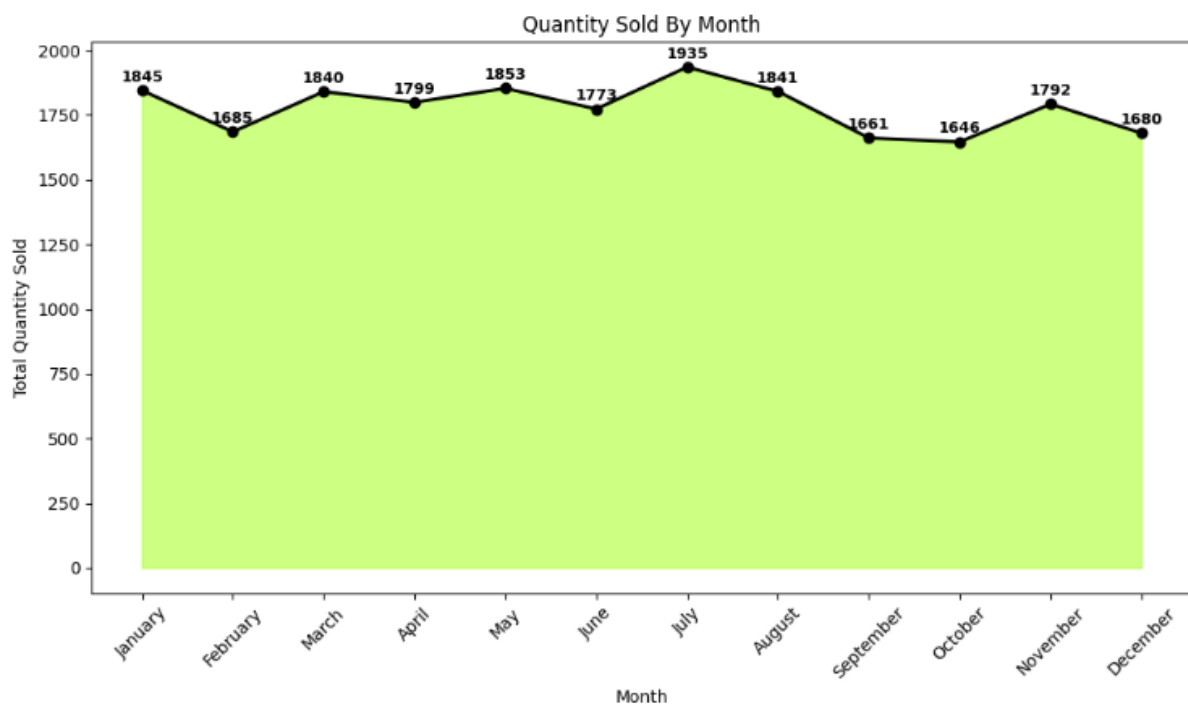
```
df["order_date"] = pd.to_datetime(df["order_date"], dayfirst=True)
df["month_name"] = df["order_date"].dt.month_name()
month_order = ["January", "February", "March", "April", "May", "June",
               "July", "August", "September", "October", "November", "December"]
df["month_name"] = pd.Categorical(df["month_name"], categories=month_order, ordered=True)
quantity_sold_by_month = df.groupby("month_name", observed=False)["quantity"].sum()

plt.figure(figsize=(10,6))
plt.fill_between(quantity_sold_by_month.index, orders_by_month.values, color="greenyellow", alpha=0.6)
plt.plot(quantity_sold_by_month.index, orders_by_month.values, color="black", linewidth=2, marker="o")

plt.title("Quantity Sold By Month")
plt.xlabel("Month")
plt.ylabel("Total Quantity Sold")
plt.xticks(rotation=45)

for i, val in enumerate(orders_by_month):
    plt.text(i, val + 20, str(val), ha="center", va="bottom", fontsize=9, fontweight="bold")

plt.tight_layout()
plt.show()
```



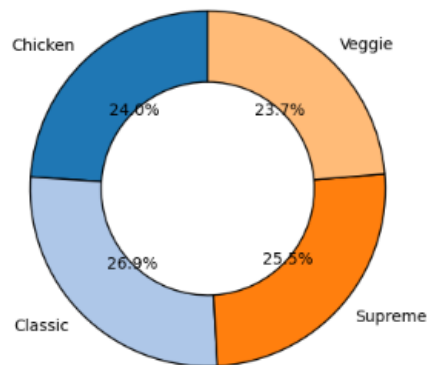
## % of Sales By Pizza Category

```
category_sales = df.groupby("pizza_category")["total_price"].sum()
category_pct = category_sales/category_sales.sum()*100

plt.figure(figsize=(5,5))
color = plt.get_cmap("tab20").colors

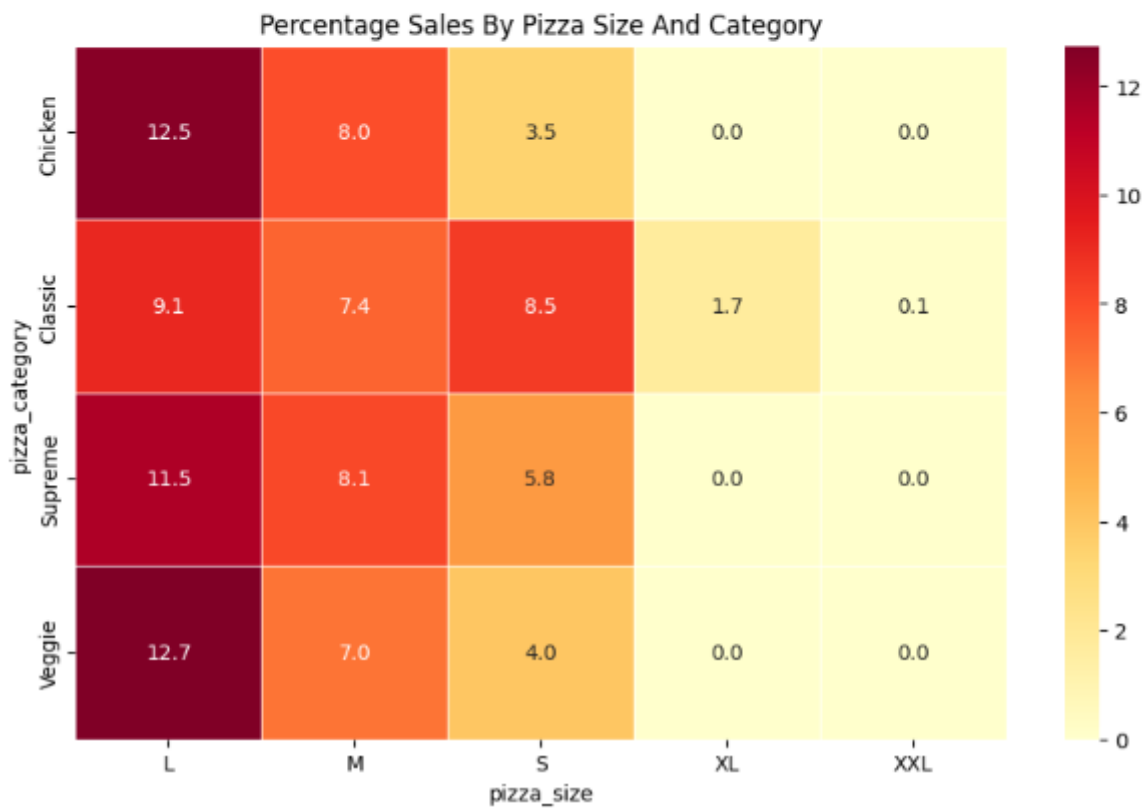
plt.pie(category_pct, labels=category_pct.index, autopct="%1.1f%%", startangle=90, colors=color, wedgeprops={"edgecolor":"black", "width":0.4})
plt.title("Percentage Of Sales By Pizza Category")
plt.show()
```

Percentage Of Sales By Pizza Category



## % Sales by Pizza Size & Category

```
sales_pivot = df.pivot_table(  
    index = "pizza_category",  
    columns = "pizza_size",  
    values = "total_price",  
    aggfunc = "sum",  
    fill_value = 0  
)  
  
sales_pct = sales_pivot/sales_pivot.sum().sum()*100  
  
plt.figure(figsize=(10,6))  
sns.heatmap(sales_pct, annot=True, fmt=".1f", cmap="YlOrRd", linewidth=0.5)  
plt.title("Percentage Sales By Pizza Size And Category")  
plt.ylabel("pizza_category")  
plt.xlabel("pizza_size")  
plt.show()
```



## Quantity Sold By Pizza Category

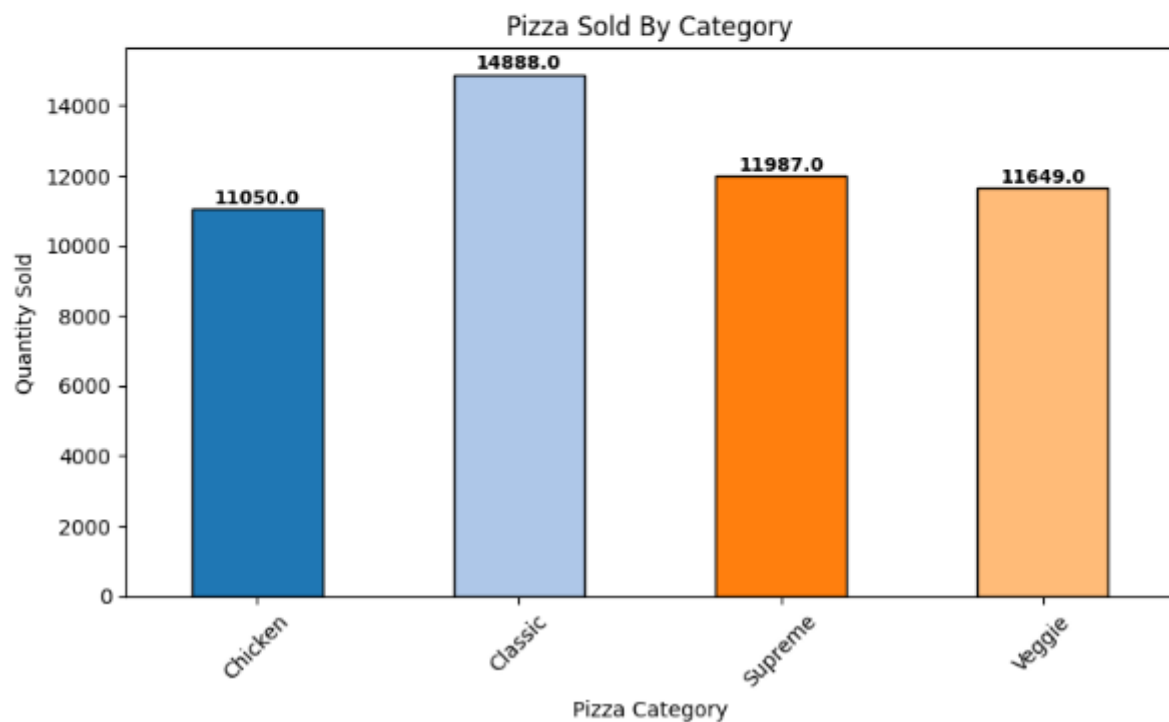
```
pizza_sold_by_category = df.groupby("pizza_category")["quantity"].sum()
color = list(plt.get_cmap("tab20").colors)
colors = color[:len(pizza_sold_by_category)]

ax = pizza_sold_by_category.plot(kind="bar", figsize=(8,5), color=colors, edgecolor="black")

plt.title("Pizza Sold By Category")
plt.xlabel("Pizza Category")
plt.ylabel("Quantity Sold")
plt.xticks(rotation=45)

for i, val in enumerate(pizza_sold_by_category):
    plt.text(i, val+20, str(val), ha="center", va="bottom", fontsize=9, fontweight="bold")

plt.tight_layout()
plt.show()
```



## Top 5 Pizzas By Quantity Sold

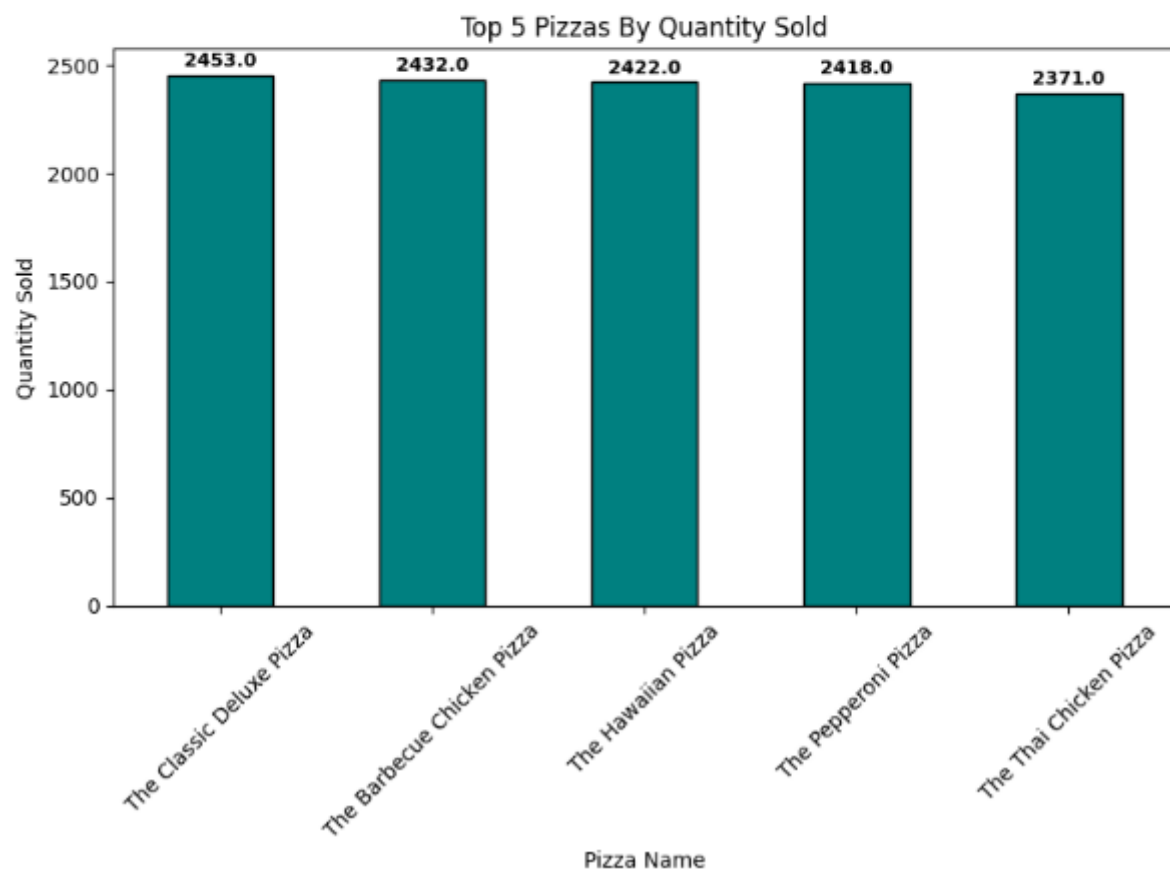
```
pizza_by_name = df.groupby("pizza_name")["quantity"].sum()
top_5 = pizza_by_name.sort_values(ascending=False).head(5)

ax = top_5.plot(kind="bar", figsize=(8,6), color="teal", edgecolor="black")

plt.title("Top 5 Pizzas By Quantity Sold")
plt.xlabel("Pizza Name")
plt.ylabel("Quantity Sold")
plt.xticks(rotation=45)

for i, val in enumerate(top_5):
    plt.text(i, val+20, str(val), ha="center", va="bottom", fontsize=9, fontweight="bold")

plt.tight_layout()
plt.show()
```



## Top 5 Pizzas By Total Revenue

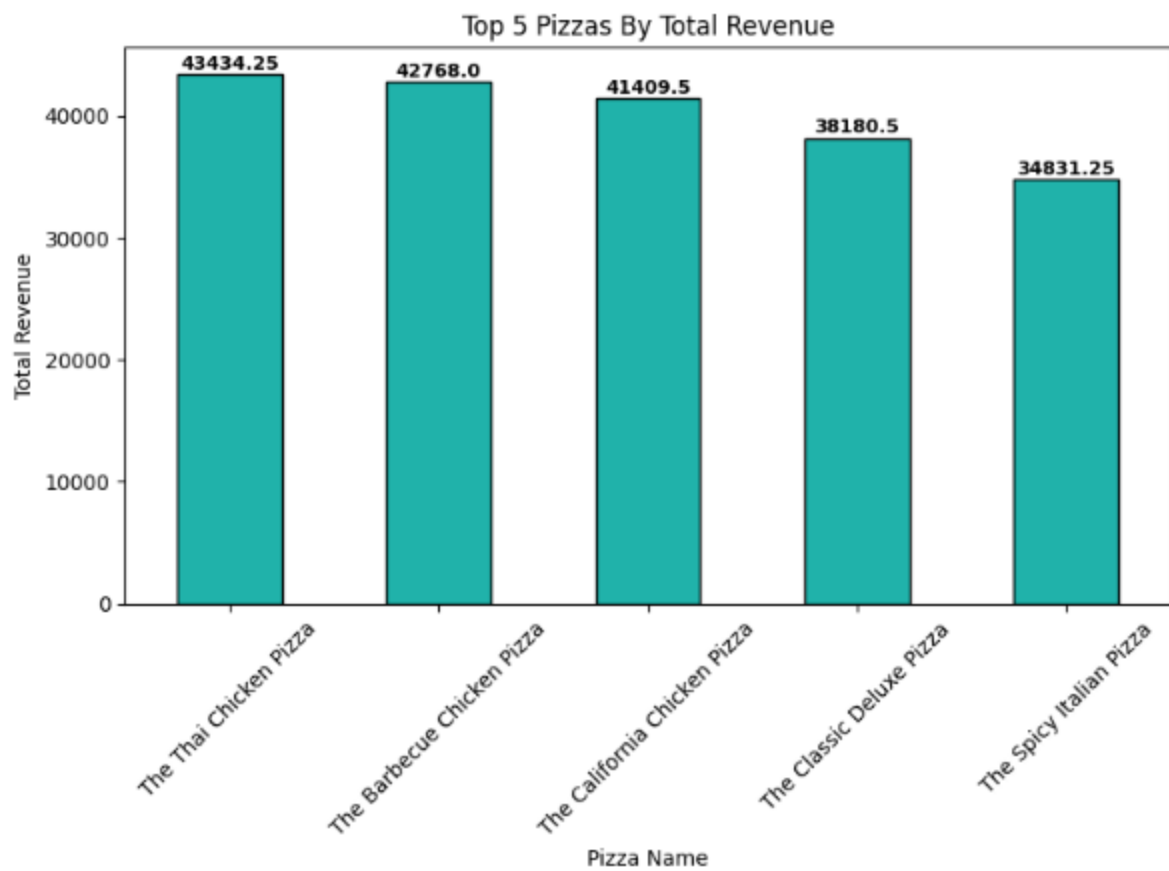
```
: pizza_by_name = df.groupby("pizza_name")["total_price"].sum()
top_5 = pizza_by_name.sort_values(ascending=False).head(5)

ax = top_5.plot(kind="bar", figsize=(8,6), color="lightseagreen", edgecolor="black")

plt.title("Top 5 Pizzas By Total Revenue")
plt.xlabel("Pizza Name")
plt.ylabel("Total Revenue")
plt.xticks(rotation=45)

for i, val in enumerate(top_5):
    plt.text(i, val+20, str(val), ha="center", va="bottom", fontsize=9, fontweight="bold")

plt.tight_layout()
plt.show()
```



## Top 5 Pizzas By Total Orders

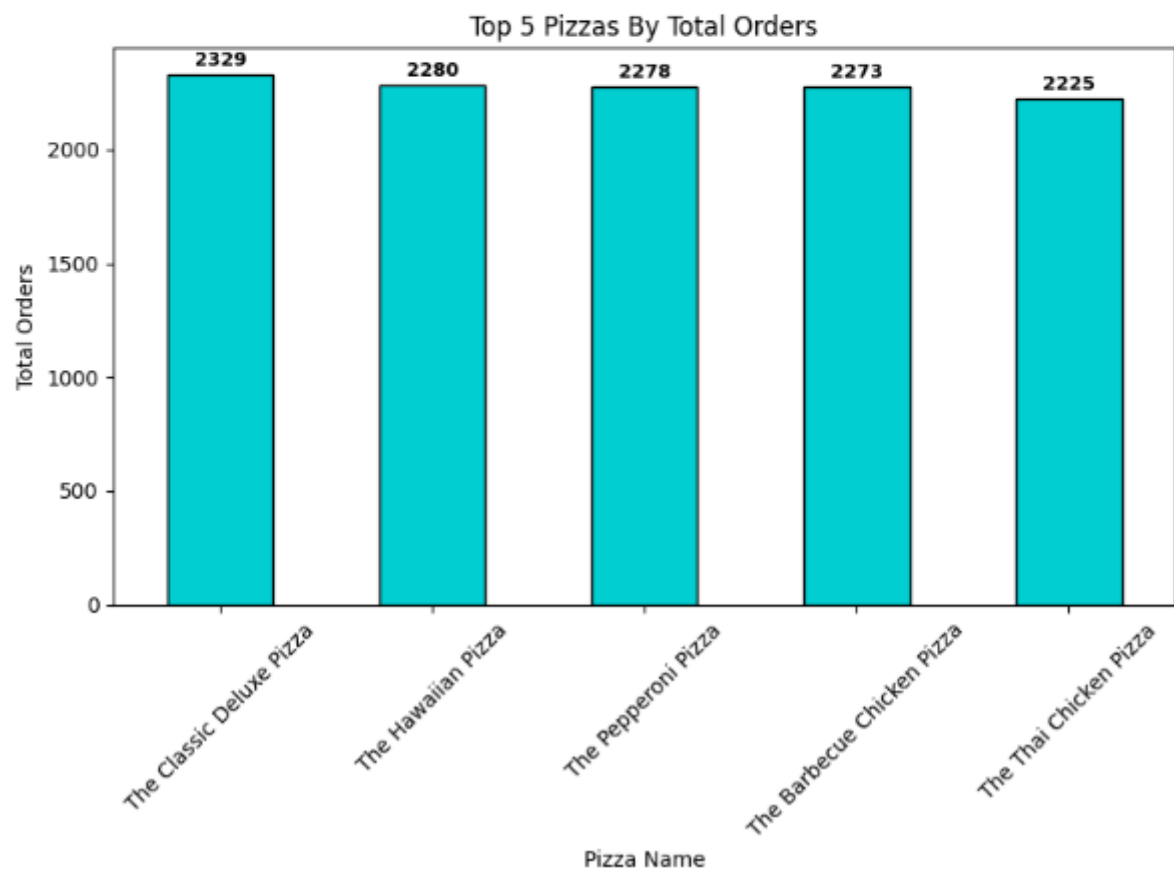
```
pizza_by_name = df.groupby("pizza_name")["order_id"].nunique()
top_5 = pizza_by_name.sort_values(ascending=False).head(5)

ax = top_5.plot(kind="bar", figsize=(8,6), color="darkturquoise", edgecolor="black")

plt.title("Top 5 Pizzas By Total Orders")
plt.xlabel("Pizza Name")
plt.ylabel("Total Orders")
plt.xticks(rotation=45)

for i, val in enumerate(top_5):
    plt.text(i, val+20, str(val), ha="center", va="bottom", fontsize=9, fontweight="bold")

plt.tight_layout()
plt.show()
```





## Bottom 5 Pizzas By Total Orders

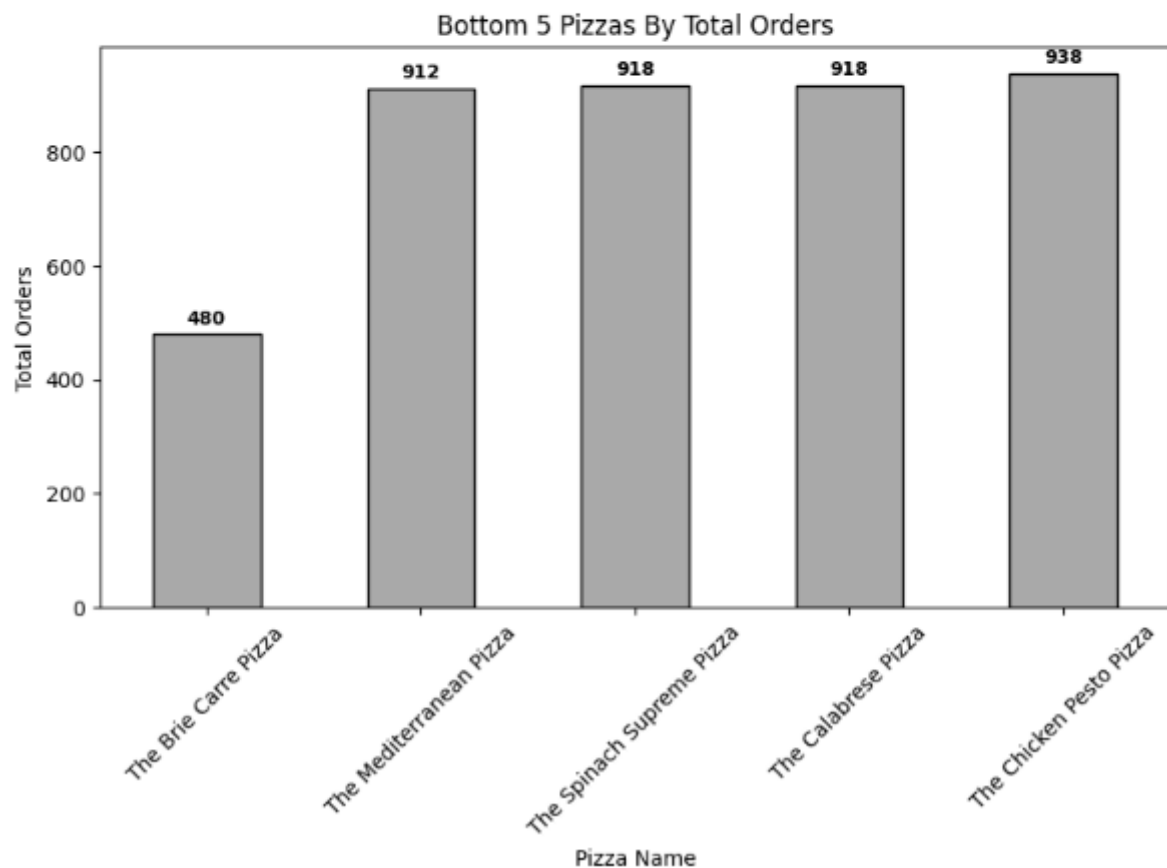
```
pizza_by_name = df.groupby("pizza_name")["order_id"].nunique()
bottom_5 = pizza_by_name.sort_values(ascending=True).head(5)

ax = bottom_5.plot(kind="bar", figsize=(8,6), color="darkgrey", edgecolor="black")

plt.title("Bottom 5 Pizzas By Total Orders")
plt.xlabel("Pizza Name")
plt.ylabel("Total Orders")
plt.xticks(rotation=45)

for i, val in enumerate(bottom_5):
    plt.text(i, val+10, str(val), ha="center", va="bottom", fontsize=9, fontweight="bold")

plt.tight_layout()
plt.show()
```



## Bottom 5 Pizzas By Total Revenue

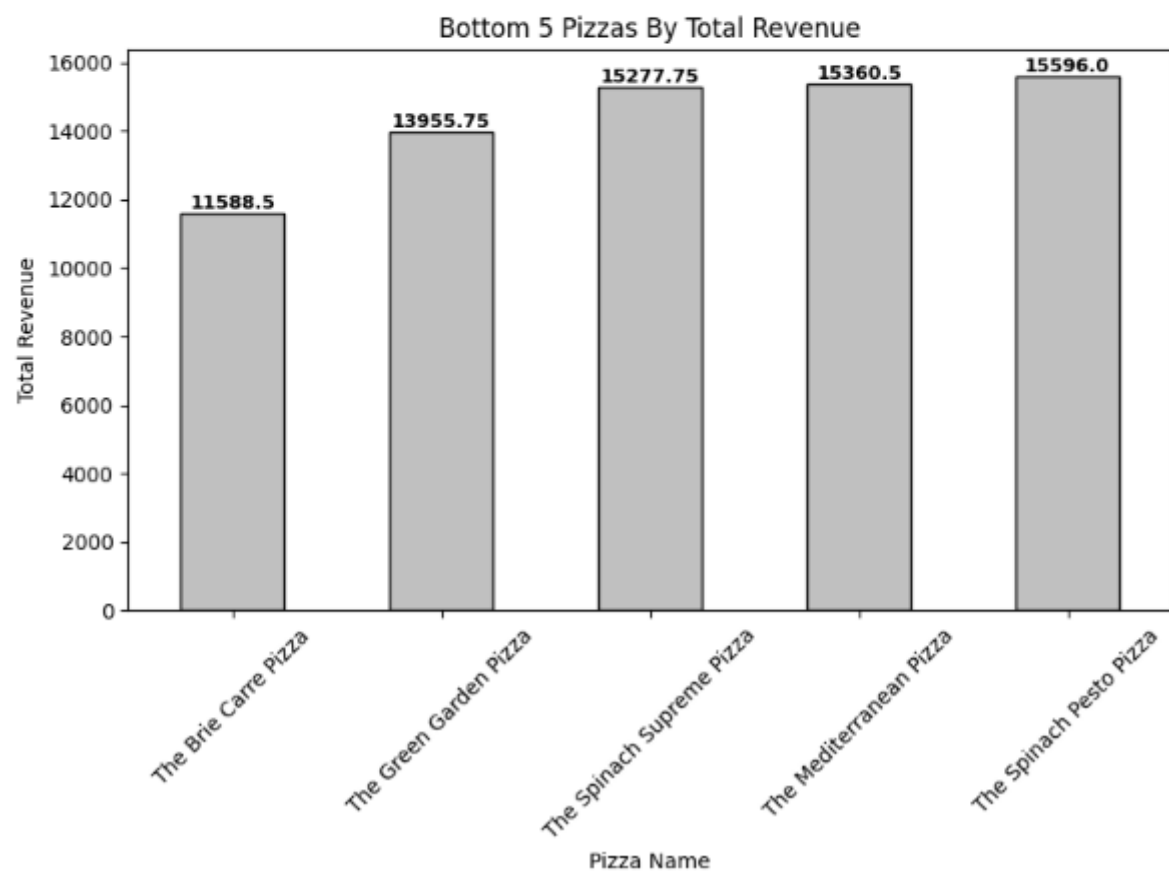
```
pizza_by_name = df.groupby("pizza_name")["total_price"].sum()
bottom_5 = pizza_by_name.sort_values(ascending=True).head(5)

ax = bottom_5.plot(kind="bar", figsize=(8,6), color="silver", edgecolor="black")

plt.title("Bottom 5 Pizzas By Total Revenue")
plt.xlabel("Pizza Name")
plt.ylabel("Total Revenue")
plt.xticks(rotation=45)

for i, val in enumerate(bottom_5):
    plt.text(i, val+10, str(val), ha="center", va="bottom", fontsize=9, fontweight="bold")

plt.tight_layout()
plt.show()
```



## Bottom 5 Pizzas By Quantity Sold

```
pizza_by_name = df.groupby("pizza_name")["quantity"].sum()
bottom_5 = pizza_by_name.sort_values(ascending=True).head(5)

ax = bottom_5.plot(kind="bar", figsize=(8,6), color="lightgrey", edgecolor="black")

plt.title("Bottom 5 Pizzas By Quantity Sold")
plt.xlabel("Pizza Name")
plt.ylabel("Quantity Sold")
plt.xticks(rotation=45)

for i, val in enumerate(bottom_5):
    plt.text(i, val+10, str(val), ha="center", va="bottom", fontsize=9, fontweight="bold")

plt.tight_layout()
plt.show()
```

