Exploratory Data Analysis (EDA) – Visual Insights Report

Tools Used: Python (Pandas, Matplotlib, Seaborn)

Datasets:

- Netflix Movies and TV Shows
- Iris Dataset

1. Dataset 1: Netflix Movies and TV Shows

Overview:

The Netflix dataset contains details about movies and TV shows including title, type, director, cast, country, release year, rating, duration, and genre.

Distribution of Numerical Features:

- **Release Year:** Most content released after 2015; spike in content after 2018; very few titles before 2000.
- **Duration:** Movies mostly 80–120 minutes; TV shows 1–3 seasons.

Categorical Feature Analysis:

- **Content Type:** 70% movies, 30% TV shows.
- **Rating Distribution:** TV-MA, TV-14, PG-13 most common.
- **Country-wise Content:** USA produces the most, followed by India, UK, and Canada.

Outlier Detection (Box Plot):

- Few movies exceed 200 minutes; most between 90–120 minutes.

Correlation Heatmap:

- Weak correlation between numerical features.
- No strong relationships observed.

Important Features for Prediction:

- Release Year, Duration, Genre, Country, Rating.

In [2]:
```
import pandas as pd
import numpy as py
import matplotlib.pyplot as plt
import seaborn as sns

df = pd.read_csv(r"C:\Users\srimullai\Downloads\archive (6)\netflix_titles.csv")
```

In [3]:
```
df.head()
```

Out[3]:

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	Documentaries	As her father nears the end of his life, filmm...
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	International TV Shows, TV Dramas, TV Mysteries	After crossing paths at a party, a Cape Town t...
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV-MA	1 Season	Crime TV Shows, International TV Shows, TV Act...	To protect his family from a powerful drug lor...
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	TV-MA	1 Season	Docuseries, Reality TV	Feuds, flirtations and toilet talk go down amo...
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA	2 Seasons	International TV Shows, Romantic TV Shows, TV ...	In a city of coaching centers known to train I...

In [4]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8807 entries, 0 to 8806
Data columns (total 12 columns):
 #   Column        Non-Null Count  Dtype 
---  ------        --------------  ----- 
 0   show_id       8807 non-null   object
 1   type          8807 non-null   object
 2   title         8807 non-null   object
 3   director      6173 non-null   object
 4   cast          7982 non-null   object
 5   country       7976 non-null   object
 6   date_added    8797 non-null   object
 7   release_year  8807 non-null   int64 
 8   rating        8803 non-null   object
 9   duration      8804 non-null   object
 10  listed_in     8807 non-null   object
 11  description   8807 non-null   object
dtypes: int64(1), object(11)
memory usage: 825.8+ KB
```

In [5]:
```
df.describe()
```

Out[5]:

	release_year
count	8807.000000
mean	2014.180198
std	8.819312
min	1925.000000
25%	2013.000000
50%	2017.000000
75%	2019.000000
max	2021.000000

In [6]:
```
df.isnull().sum()
```

Out[6]:
```
show_id          0
type             0
title            0
director      2634
cast           825
country        831
date_added      10
release_year     0
rating           4
duration         3
listed_in        0
description      0
dtype: int64
```

In [7]:
```
df.shape
```

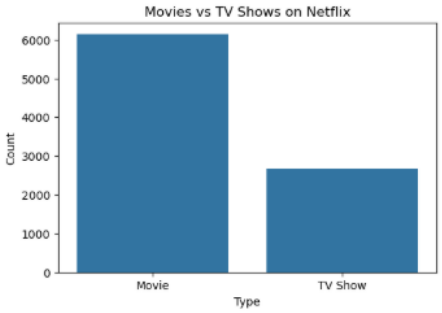
Out[7]:
```
(8807, 12)
```

In [8]:
```
df.tail()
```

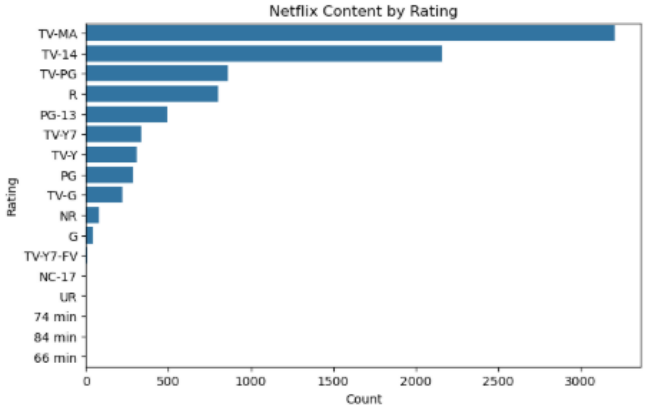
Out[8]:

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
8802	s8803	Movie	Zodiac	David Fincher	Mark Ruffalo, Jake Gyllenhaal, Robert Downey J...	United States	November 20, 2019	2007	R	158 min	Cult Movies, Dramas, Thrillers	A political cartoonist, a crime reporter and a...
8803	s8804	TV Show	Zombie Dumb	NaN	NaN	NaN	July 1, 2019	2018	TV-Y7	2 Seasons	Kids' TV, Korean TV Shows, TV Comedies	While living alone in a spooky town, a young g...
8804	s8805	Movie	Zombieland	Ruben Fleischer	Jesse Eisenberg, Woody Harrelson, Emma Stone, ...	United States	November 1, 2019	2009	R	88 min	Comedies, Horror Movies	Looking to survive in a world taken over by zo...
8805	s8806	Movie	Zoom	Peter Hewitt	Tim Allen, Courteney Cox, Chevy Chase, Kate Ma...	United States	January 11, 2020	2006	PG	88 min	Children & Family Movies, Comedies	Dragged from civilian life, a former superhero...
8806	s8807	Movie	Zubaan	Mozez Singh	Vicky Kaushal, Sarah-Jane Dias, Raaghav Chanan...	India	March 2, 2019	2015	TV-14	111 min	Dramas, International Movies, Music & Musicals	A scrappy but poor boy worms his way into a ty...

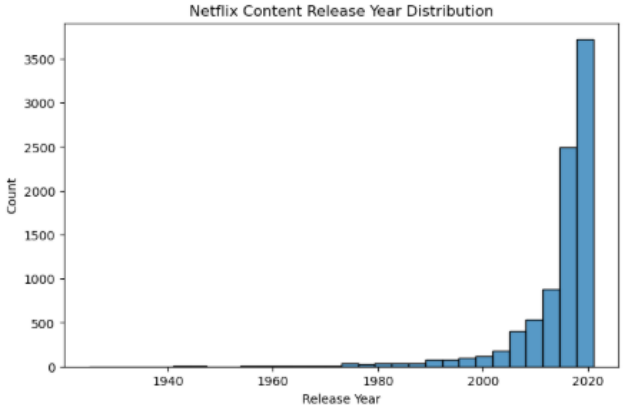
In [9]:
```
plt.figure(figsize=(6,4))
sns.countplot(x='type', data=df)
plt.title("Movies vs TV Shows on Netflix")
plt.xlabel("Type")
plt.ylabel("Count")
plt.show()
```



In [10]:
```
plt.figure(figsize=(8,5))
sns.countplot(y='rating', data=df, order=df['rating'].value_counts().index)
plt.title("Netflix Content by Rating")
plt.xlabel("Count")
plt.ylabel("Rating")
plt.show()
```

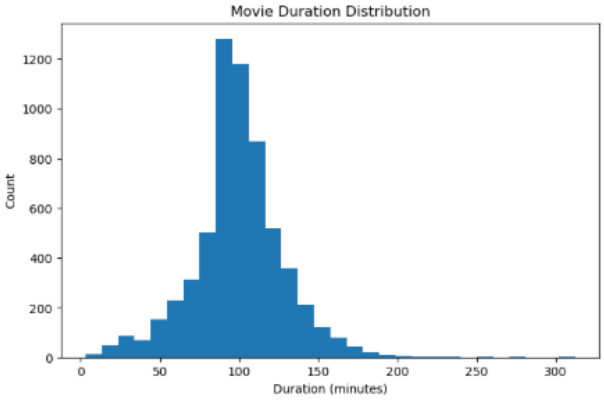


In [11]:
```
plt.figure(figsize=(8,5))
sns.histplot(df['release_year'], bins=30)
plt.title("Netflix Content Release Year Distribution")
plt.xlabel("Release Year")
plt.ylabel("Count")
plt.show()
```



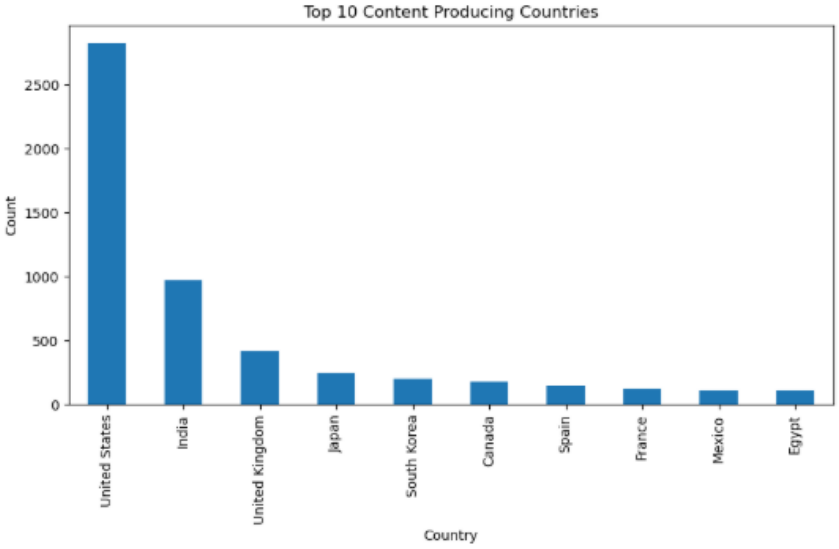
In [12]:
```
movies = df[df['type'] == 'Movie'].copy()
movies['duration'] = movies['duration'].str.replace(' min','')
movies['duration'] = pd.to_numeric(movies['duration'])

plt.figure(figsize=(8,5))
plt.hist(movies['duration'], bins=30)
plt.title("Movie Duration Distribution")
plt.xlabel("Duration (minutes)")
plt.ylabel("Count")
plt.show()
```



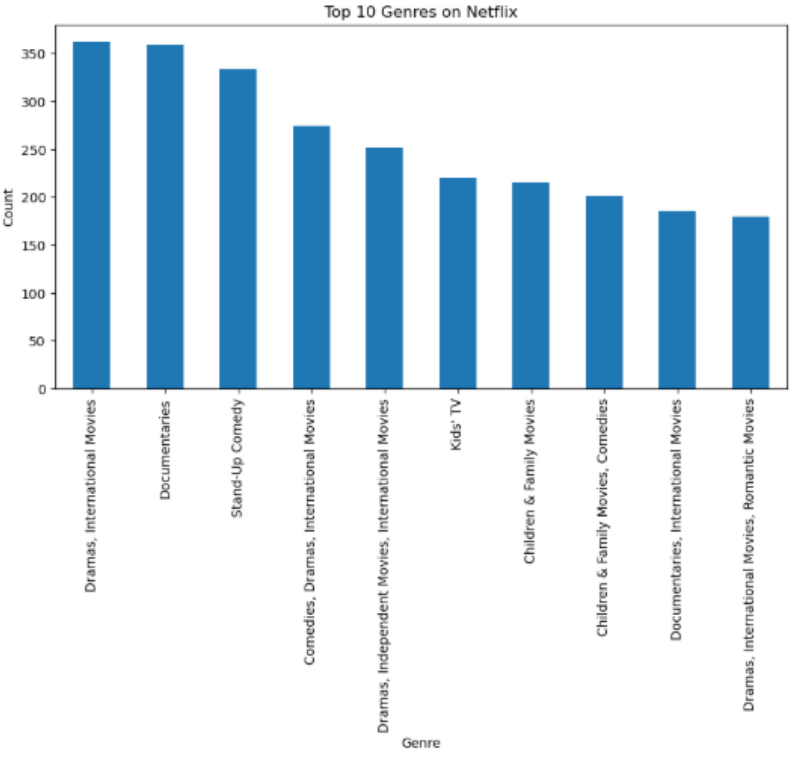
In [13]:

```
plt.figure(figsize=(10,5))
df['country'].value_counts().head(10).plot(kind='bar')
plt.title("Top 10 Content Producing Countries")
plt.xlabel("Country")
plt.ylabel("Count")
plt.show()
```

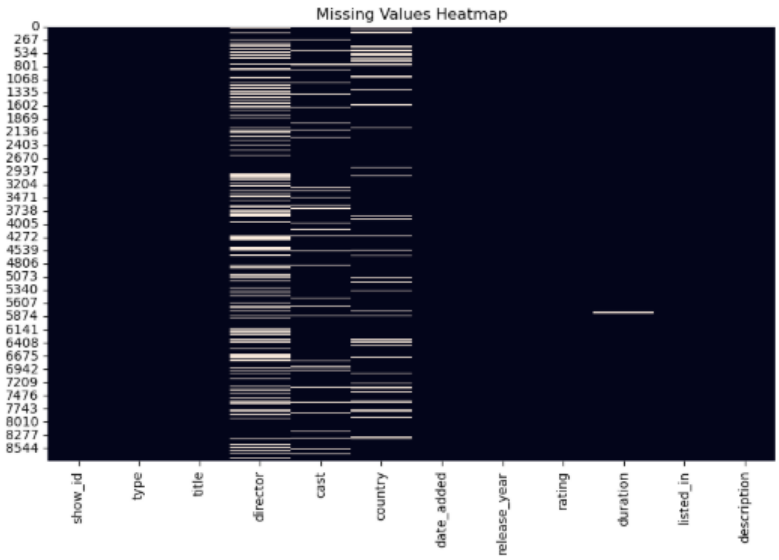


In [14]:

```
df['listed_in'].value_counts().head(10).plot(kind='bar', figsize=(10,5))
plt.title("Top 10 Genres on Netflix")
plt.xlabel("Genre")
plt.ylabel("Count")
plt.show()
```



In [15]:
```
plt.figure(figsize=(10,6))
sns.heatmap(df.isnull(), cbar=False)
plt.title("Missing Values Heatmap")
plt.show()
```



Conclusion:

- Netflix hosts more movies than TV shows.
- Majority of content is recent (post-2015).
- Teen and adult content dominates.
- USA and India are top content producers.
- Most movies 90–120 minutes; few extremely long movies exist.

2. Dataset 2: Iris Dataset

Overview:

Contains 150 flower samples with 4 features: Sepal Length, Sepal Width, Petal Length, Petal Width, and Species (Setosa, Versicolor, Virginica). Each species has 50 samples.

Distribution of Numerical Features:

- **Sepal Length:** 4.3–7.9 cm; most 5.5–6.5 cm.
- **Sepal Width:** 2.0–4.4 cm; most 2.5–3.5 cm.
- **Petal Length & Width:** Clear separation of species; Setosa has smallest petals.

Categorical Feature Analysis:

- Balanced dataset: 50 samples per species.

Outlier Detection (Box Plot):

- Sepal Width shows few outliers; petal features mostly clean.

Correlation Heatmap:

- Petal Length vs Petal Width: very strong positive correlation
- Sepal Length vs Petal Length: strong positive correlation
- Sepal Width vs Petal Width: weak negative correlation

Important Features for Prediction:

- Petal Length, Petal Width, Sepal Length
- Sepal Width less significant.

In [2]:
```
import pandas as pd
import numpy as py
import matplotlib.pyplot as plt
import seaborn as sns

df = pd.read_csv(r"C:\Users\srimullai\Downloads\archive (5)\IRIS.csv")
```

In [3]:
```
df.head()
```

Out[3]:

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa

In [4]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 5 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   sepal_length  150 non-null    float64
 1   sepal_width   150 non-null    float64
 2   petal_length  150 non-null    float64
 3   petal_width   150 non-null    float64
 4   species       150 non-null    object 
dtypes: float64(4), object(1)
memory usage: 6.0+ KB
```

In [5]:
```
df.describe()
```

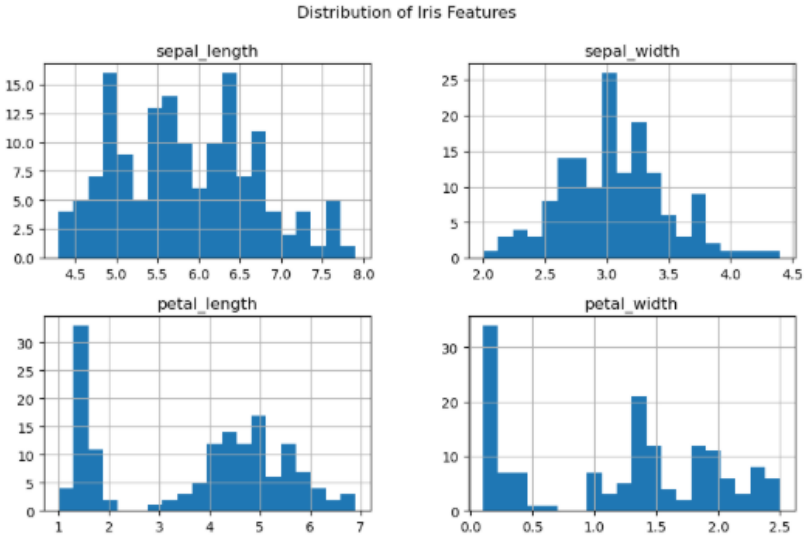
Out[5]:

	sepal length	sepal width	petal length	petal width
count	150.000000	150.000000	150.000000	150.000000
mean	5.843333	3.054000	3.758667	1.198667
std	0.828066	0.433594	1.764420	0.763161
min	4.300000	2.000000	1.000000	0.100000
25%	5.100000	2.800000	1.600000	0.300000
50%	5.800000	3.000000	4.350000	1.300000
75%	6.400000	3.300000	5.100000	1.800000
max	7.900000	4.400000	6.900000	2.500000

In [8]:
```
df.shape
```

Out[8]: (150, 5)

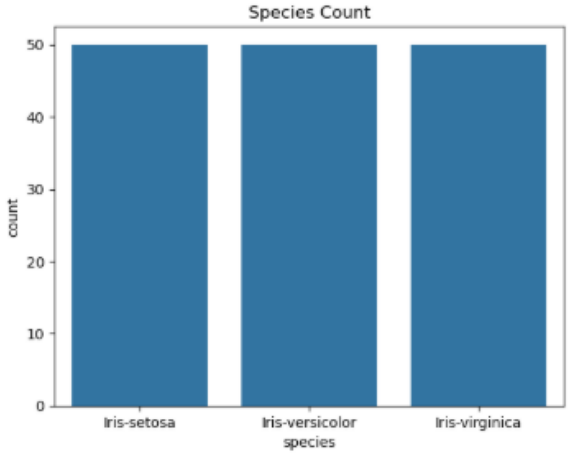
In [11]:
```
df.hist(figsize=(10,6), bins=20)
plt.suptitle("Distribution of Iris Features")
plt.show()
```



In [13]:

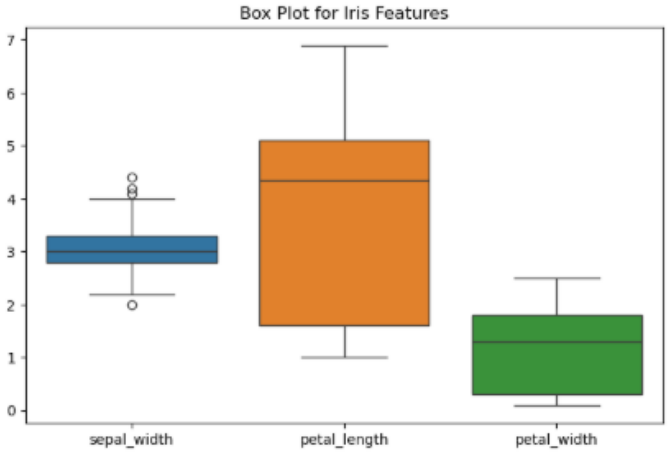
```
print(df.columns)
sns.countplot(x='species', data=df)
plt.title("Species Count")
plt.show()
```

```
Index(['sepal_length', 'sepal_width', 'petal_length', 'petal_width',
       'species'],
      dtype='object')
```

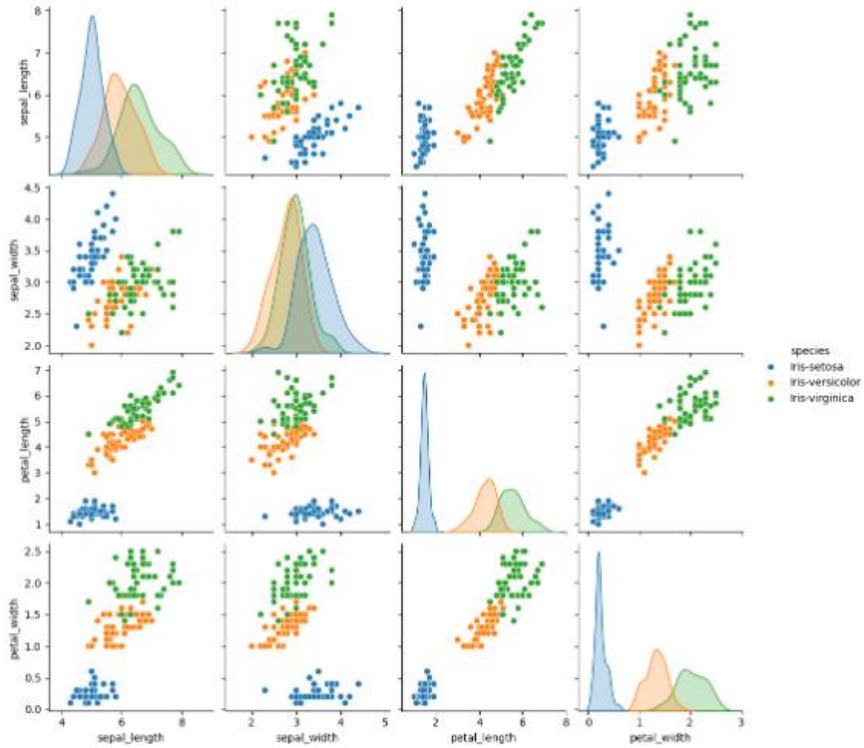


In [14]:

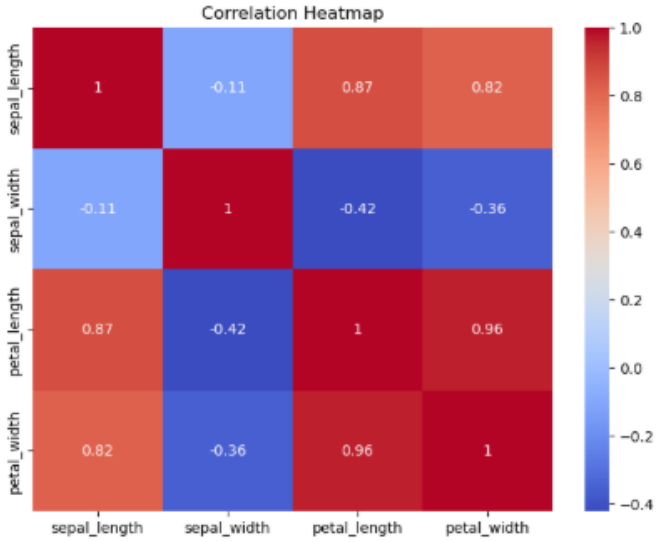
```
plt.figure(figsize=(8,5))
sns.boxplot(data=df.iloc[:,1:5])
plt.title("Box Plot for Iris Features")
plt.show()
```



In [19]:
```
sns.pairplot(df, hue="species")
plt.show()
```



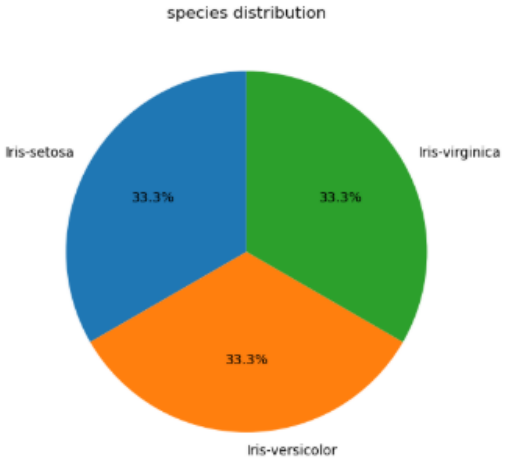
In [17]:
```
numerical_cols = df.select_dtypes(include=['float64', 'int64']).columns
plt.figure(figsize=(8,6))
sns.heatmap(df[numerical_cols].corr(), annot=True, cmap="coolwarm")
plt.title("Correlation Heatmap")
plt.show()
```



In [20]:
```
df.isnull().sum()
```

Out[20]:
```
sepal_length    0
sepal_width     0
petal_length    0
petal_width     0
species         0
dtype: int64
```

In [22]:
```
plt.figure(figsize=(6,6))
df['species'].value_counts().plot.pie(autopct='%1.1f%%', startangle=90)
plt.title("species distribution")
plt.ylabel(' ')
plt.show()
```



Conclusion:

- Clean, balanced dataset
- Petal features clearly separate species
- Strong correlation between petal length and width
- Setosa easiest to classify
- Ideal for ML classification

NAME: S.SRI MULLAI HARINIE