Project Business Statistics: E-news Express

Marks: 60

Business Context

The advent of e-news, or electronic news, portals has offered us a great opportunity to quickly get updates on the day-to-day events occurring globally. The information on these portals is retrieved electronically from online databases, processed using a variety of software, and then transmitted to the users. There are multiple advantages of transmitting new electronically, like faster access to the content and the ability to utilize different technologies such as audio, graphics, video, and other interactive elements that are either not being used or aren't common yet in traditional newspapers.

E-news Express, an online news portal, aims to expand its business by acquiring new subscribers. With every visitor to the website taking certain actions based on their interest, the company plans to analyze these actions to understand user interests and determine how to drive better engagement. The executives at E-news Express are of the opinion that there has been a decline in new monthly subscribers compared to the past year because the current webpage is not designed well enough in terms of the outline & recommended content to keep customers engaged long enough to make a decision to subscribe.

[Companies often analyze user responses to two variants of a product to decide which of the two variants is more effective. This experimental technique, known as A/B testing, is used to determine whether a new feature attracts users based on a chosen metric.]

Objective

The design team of the company has researched and created a new landing page that has a new outline & more relevant content shown compared to the old page. In order to test the effectiveness of the new landing page in gathering new subscribers, the Data Science team conducted an experiment by randomly selecting 100 users and dividing them equally into two groups. The existing landing page was served to the first group (control group) and the new landing page to the second group (treatment group). Data regarding the interaction of users in both groups with the

two versions of the landing page was collected. Being a data scientist in E-news Express, you have been asked to explore the data and perform a statistical analysis (at a significance level of 5%) to determine the effectiveness of the new landing page in gathering new subscribers for the news portal by answering the following questions:

1. Do the users spend more time on the new landing page than on the existing landing page?
2. Is the conversion rate (the proportion of users who visit the landing page and get converted) for the new page greater than the conversion rate for the old page?
3. Does the converted status depend on the preferred language? [Hint: Create a contingency table using the pandas.crosstab() function]
4. Is the time spent on the new page the same for the different language users?

Data Dictionary

The data contains information regarding the interaction of users in both groups with the two versions of the landing page.

1. user_id - Unique user ID of the person visiting the website
2. group - Whether the user belongs to the first group (control) or the second group (treatment)
3. landing_page - Whether the landing page is new or old
4. time_spent_on_the_page - Time (in minutes) spent by the user on the landing page
5. converted - Whether the user gets converted to a subscriber of the news portal or not
6. language_preferred - Language chosen by the user to view the landing page

Please read the instructions carefully before starting the project.

This is a commented Jupyter IPython Notebook file in which all the instructions and tasks to be performed are mentioned.

- Blanks '___' are provided in the notebook that needs to be filled with an appropriate code to get the correct result. With every '___' blank, there is a comment that briefly describes what needs to be filled in the blank space.
- Identify the task to be performed correctly, and only then proceed to write the required code.
- Fill the code wherever asked by the commented lines like "# write your code here" or "# complete the code". Running incomplete code may throw error.
- Please run the codes in a sequential manner from the beginning to avoid any unnecessary errors.
- Add the results/observations (wherever mentioned) derived from the analysis in the presentation and submit the same. Any mathematical or computational details which are a graded part of the project can be included in the Appendix section of the presentation.

Import all the necessary libraries

In [15]:
```
import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
import seaborn as sns
```

Loading the dataset

In [16]:
```
# complete the code below to load the dataset
df = pd.read_csv('abtest.csv')
```

Explore the dataset and extract insights using Exploratory Data Analysis

Data Overview

The initial steps to get an overview of any dataset is to:

- observe the first few rows of the dataset, to check whether the dataset has been loaded properly or not
- get information about the number of rows and columns in the dataset
- find out the data types of the columns to ensure that data is stored in the preferred format and the value of each property is as expected.
- check the statistical summary of the dataset to get an overview of the numerical columns of the data

Displaying the first few rows of the dataset

In [17]:
```
# view the first 5 rows of the dataset
df.head(5)
```

Out[17]:

	user_id	group	landing_page	time_spent_on_the_page	converted	language_prefe
0	546592	control	old	3.48	no	Sp
1	546468	treatment	new	7.13	yes	En
2	546462	treatment	new	4.40	no	Sp
3	546567	control	old	3.02	no	Fr
4	546459	treatment	new	4.75	yes	Sp

Displaying the last few rows of the dataset

In [18]:
```
# view the last 5 rows of the dataset
df.tail(5)
```

Out[18]:

	user_id	group	landing_page	time_spent_on_the_page	converted	language_pre
95	546446	treatment	new	5.15	no	S
96	546544	control	old	6.52	yes	E
97	546472	treatment	new	7.07	yes	S
98	546481	treatment	new	6.20	yes	S
99	546483	treatment	new	5.86	yes	E

Checking the shape of the dataset

In [19]:
```
# view the shape of the dataset
df.shape
```

Out[19]:
```
(100, 6)
```

Checking the data types of the columns for the dataset

In [20]:
```
# check the data types of the columns in the dataset
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 100 entries, 0 to 99
Data columns (total 6 columns):
 #   Column                  Non-Null Count  Dtype  
---  ------                  --------------  -----  
 0   user_id                 100 non-null    int64  
 1   group                   100 non-null    object 
 2   landing_page            100 non-null    object 
 3   time_spent_on_the_page  100 non-null    float64
 4   converted               100 non-null    object 
 5   language_preferred      100 non-null    object 
dtypes: float64(1), int64(1), object(4)
memory usage: 4.8+ KB
```

Getting the statistical summary for the numerical variables

In [21]:
```
# write your code here to print the numerical summary statistics
df.describe()
```

Out[21]:

	user_id	time_spent_on_the_page
count	100.000000	100.000000
mean	546517.000000	5.377800
std	52.295779	2.378166
min	546443.000000	0.190000
25%	546467.750000	3.880000
50%	546492.500000	5.415000
75%	546567.250000	7.022500
max	546592.000000	10.710000

Getting the statistical summary for the categorical variables

In [24]:
```
# write your code here to print the categorical summary statistics
df.describe(exclude="number").T
```

Out[24]:

	count	unique	top	freq
group	100	2	control	50
landing_page	100	2	old	50
converted	100	2	yes	54
language_preferred	100	3	Spanish	34

Check for missing values

In [25]:
```
# write your code here
df.isnull().sum()
```

Out[25]:
```
user_id                     0
group                       0
landing_page                0
time_spent_on_the_page      0
converted                   0
language_preferred          0
dtype: int64
```

Check for duplicates

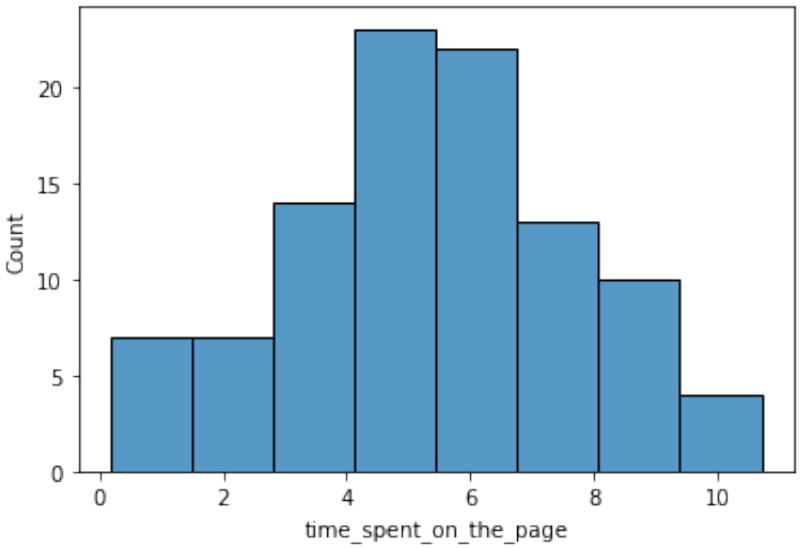
In [26]:
```
# write your code here
df.user_id.nunique()
```

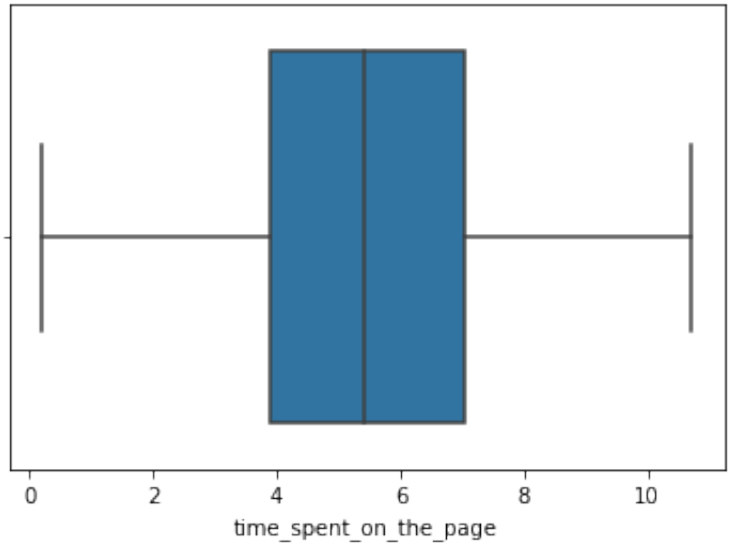
Out[26]:
```
100
```

Univariate Analysis

Time spent on the page

In [27]:
```
sns.histplot(data=df,x='time_spent_on_the_page')
plt.show()
sns.boxplot(data=df,x='time_spent_on_the_page')
plt.show()
```



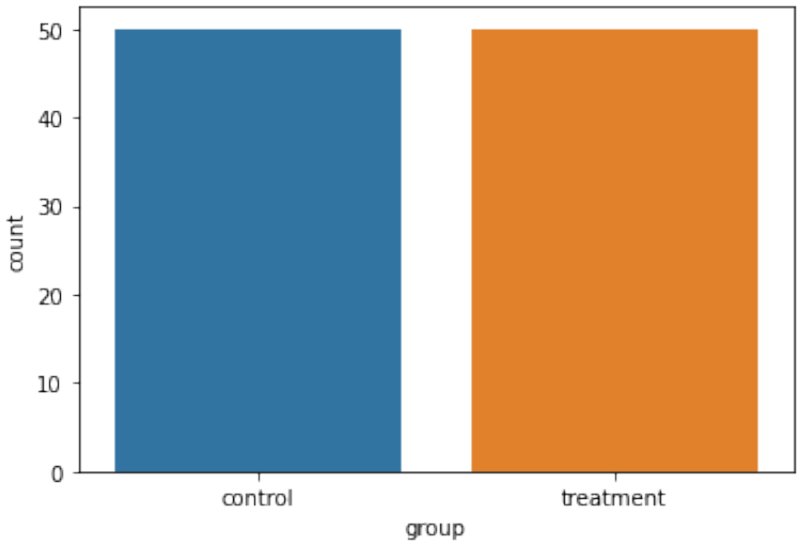


Group

In [28]:
```
df['group'].value_counts()
```

Out[28]:
```
control      50
treatment    50
Name: group, dtype: int64
```

In [29]:
```
sns.countplot(data=df,x='group')
plt.show()
```

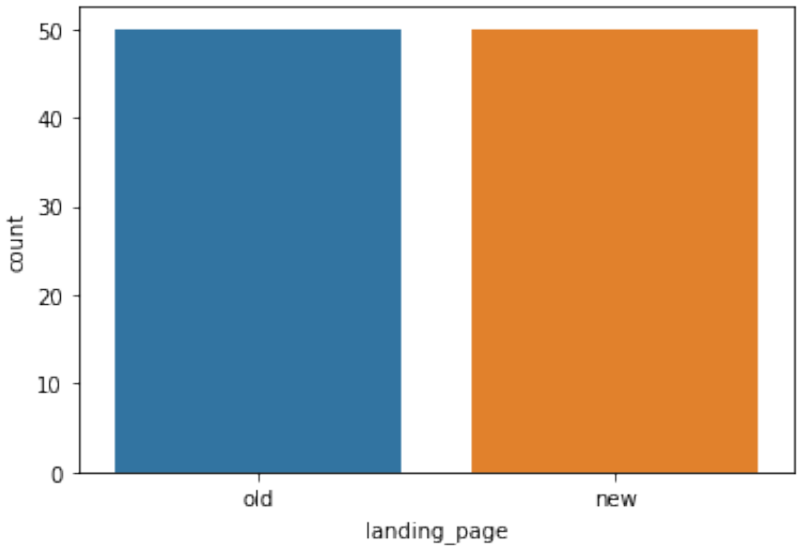


Landing page

In [30]:
```
df['landing_page'].value_counts()
```

Out[30]:
```
old    50
new    50
Name: landing_page, dtype: int64
```

In [32]:
```
# complete the code to plot the countplot
sns.countplot(data=df,x="landing_page")
plt.show()
```

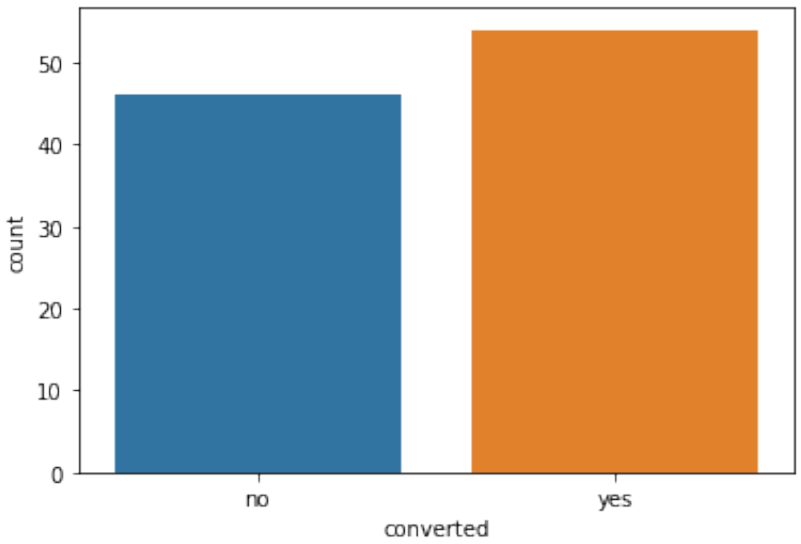


Converted

In [33]:
```
df['converted'].value_counts()
```

Out[33]:
```
yes    54
no     46
Name: converted, dtype: int64
```

In [34]:
```
# complete the code to plot the countplot
sns.countplot(data=df,x="converted")
plt.show()
```

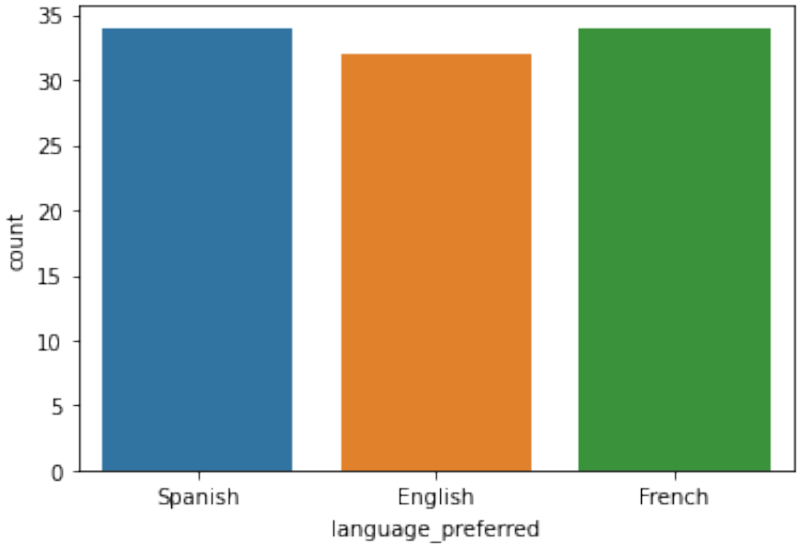


Language preferred

In [35]:
```
df['language_preferred'].value_counts()
```

Out[35]:
```
Spanish     34
French      34
English     32
Name: language_preferred, dtype: int64
```

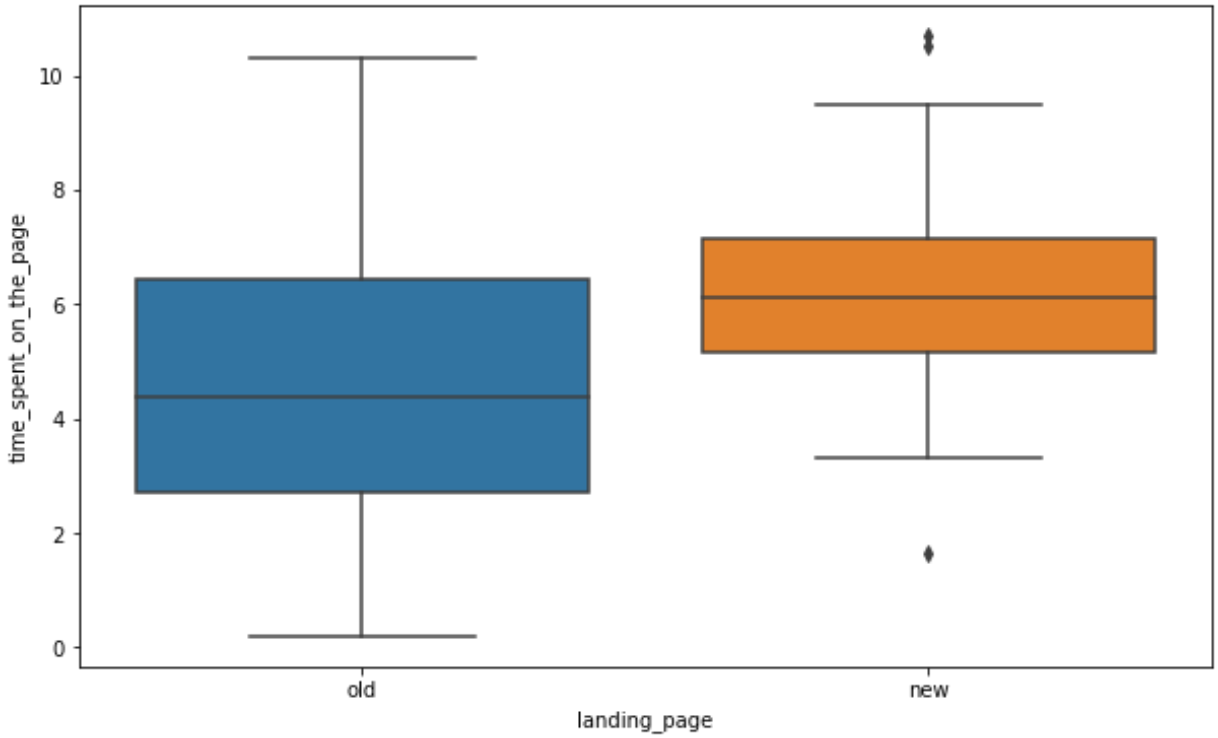
In [37]:
```
# complete the code to plot the countplot
sns.countplot(data=df,x="language_preferred")
plt.show()
```



Bivariate Analysis

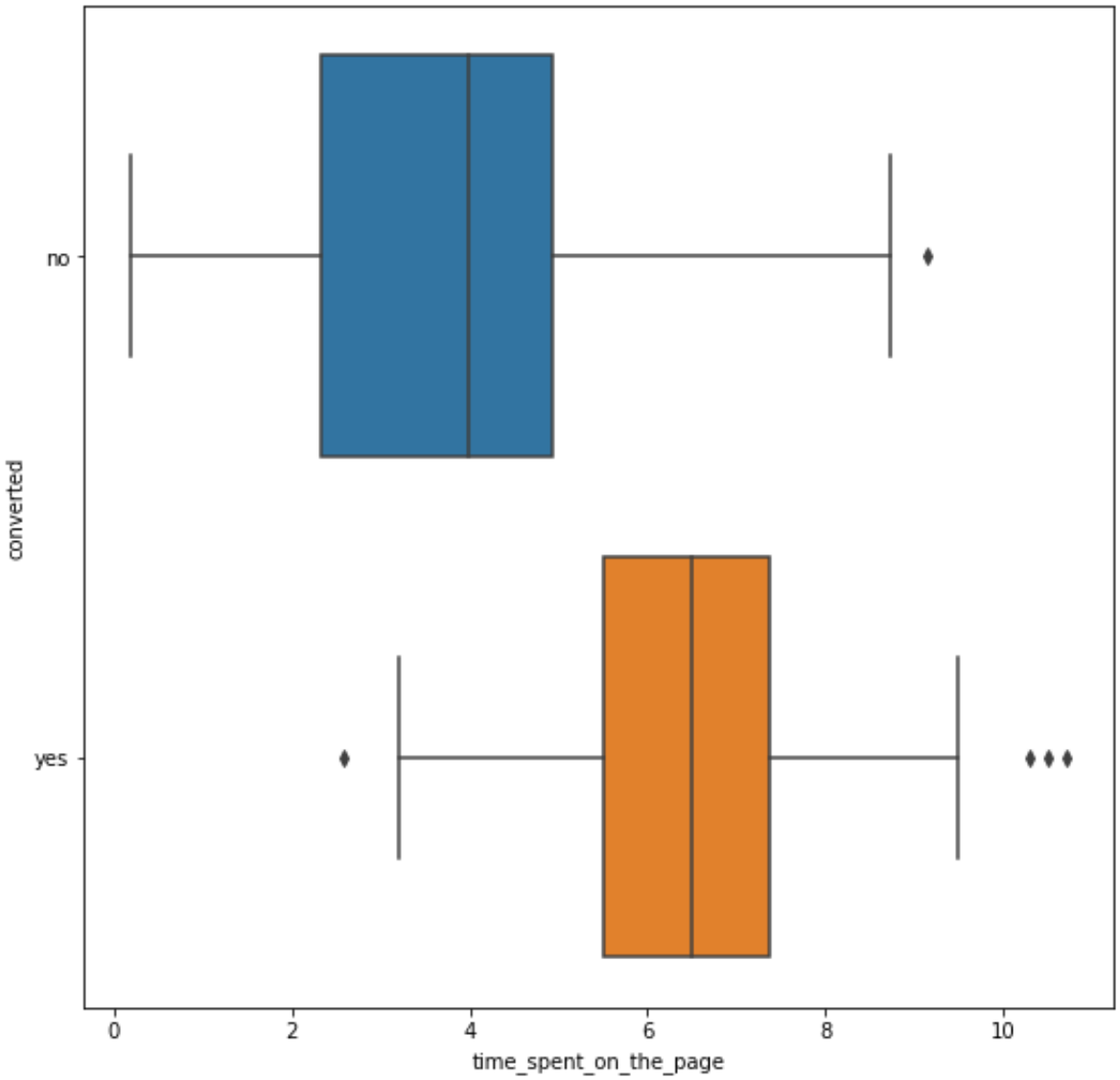
Landing page vs Time spent on the page

In [38]:
```
plt.figure(figsize=(10,6))
sns.boxplot(data=df,x='landing_page',y='time_spent_on_the_page')
plt.show()
```



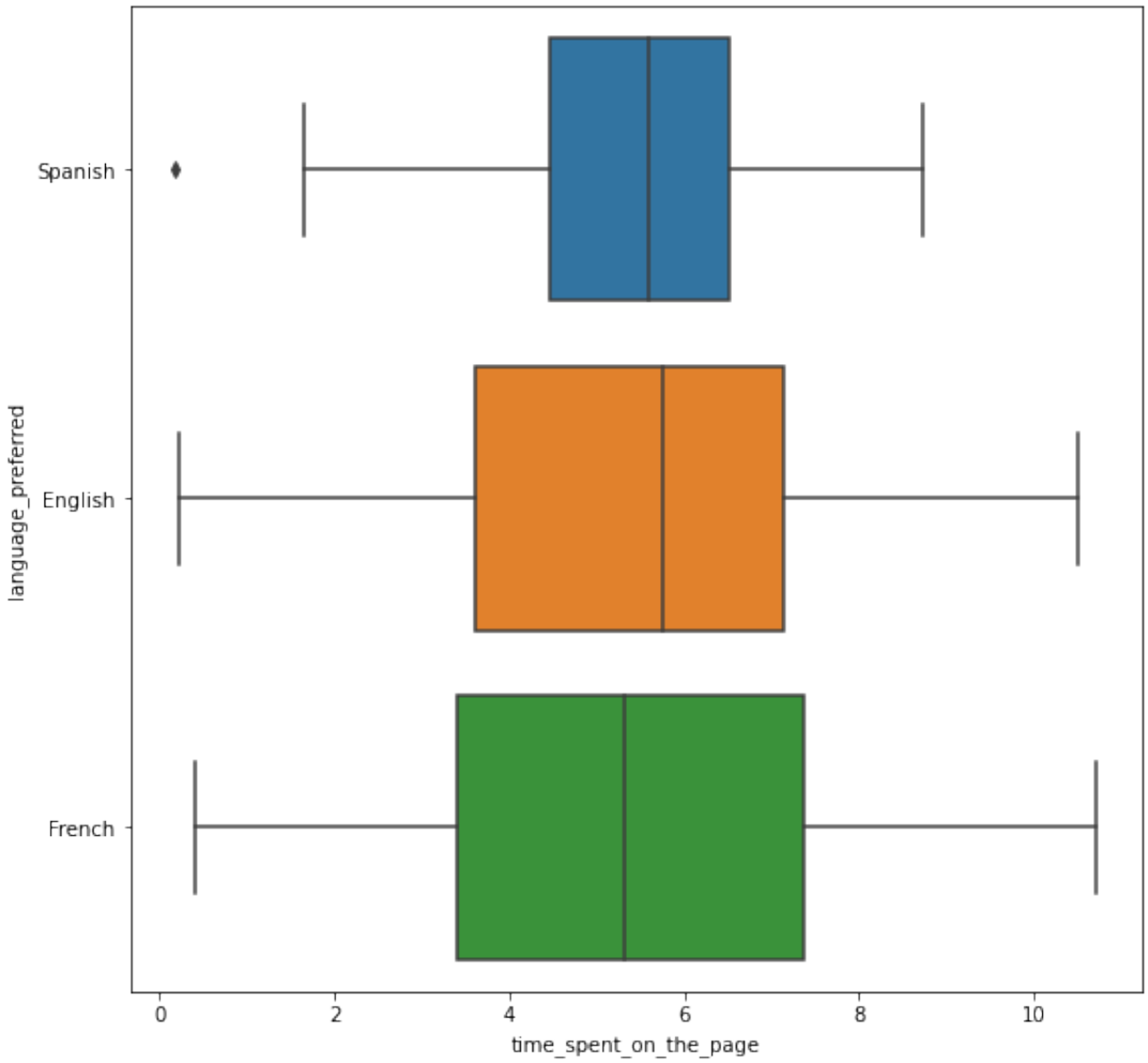
Conversion status vs Time spent on the page

In [39]:
```
# complete the code to plot a suitable graph to understand the relationsh
plt.figure(figsize=(9, 9))
sns.boxplot(data = df, x = "time_spent_on_the_page", y ='converted')
plt.show()
```



Language preferred vs Time spent on the page

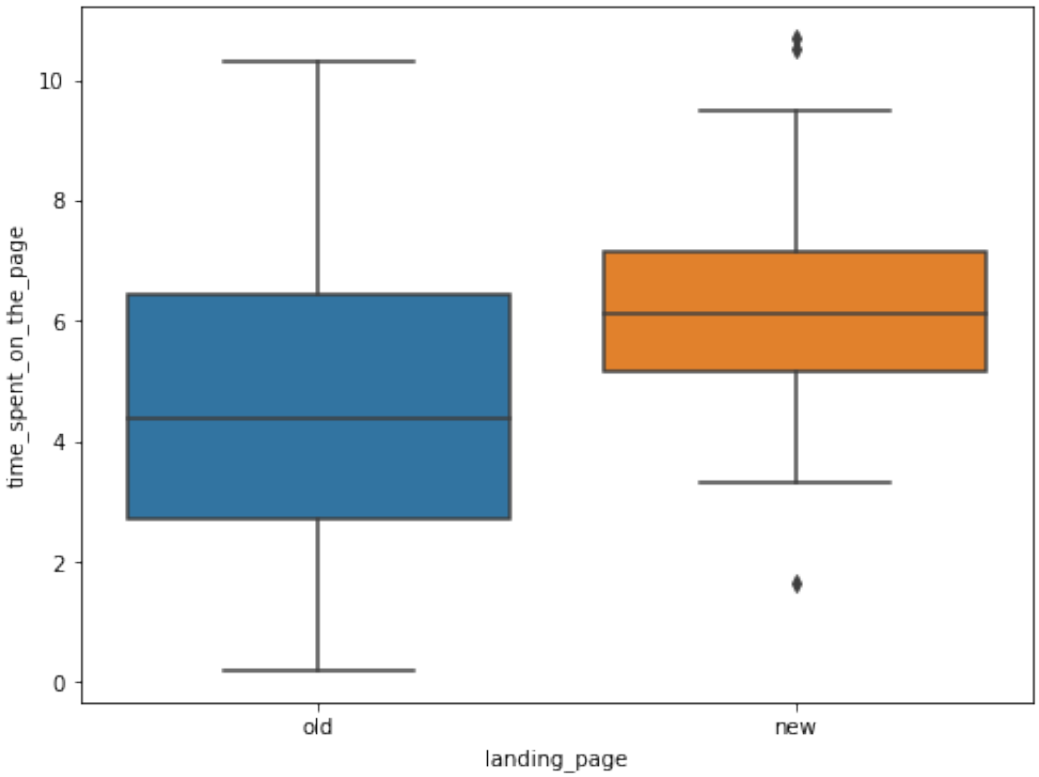
In [40]:
```
# write the code to plot a suitable graph to understand the distribution
plt.figure(figsize=(9, 9))
sns.boxplot(data = df, x = 'time_spent_on_the_page', y = 'language_prefer
plt.show()
```



1. Do the users spend more time on the new landing page than the existing landing page?

Perform Visual Analysis

In [41]:
```
# visual analysis of the time spent on the new page and the time spent on
plt.figure(figsize=(8,6))
sns.boxplot(x = 'landing_page', y = 'time_spent_on_the_page', data = df)
plt.show()
```



Step 1: Define the null and alternate hypotheses

Null: There is no difference in the time spent on the new and existing landing pages.

Alternate : There is a difference with users spending more time on the new landing page.

$H_0 : \mu_1 = \mu_2$

$H_a : \mu_1 > \mu_2$

Step 2: Select Appropriate test

(This is a one-tailed test concerning two population means from two independent populations. The population standard deviations are unknown. **Based on this information, select the appropriate test**.)

Based on the information provided, the appropriate test to use would be a Student's t-test for independent samples, also known as a two-sample independent t-test. This test is used to compare the means of two independent populations when the population standard deviations are unknown.

The t-test assumes that the two samples are independent and normally distributed, and that the variances of the two populations are equal.

Step 3: Decide the significance level

As given in the problem statement, we select $\alpha = 0.05$.

Step 4: Collect and prepare data

In [42]:
```
# create subsetted data frame for new landing page users
time_spent_new = df[df['landing_page'] == 'new']['time_spent_on_the_page'

# create subsetted data frame for old landing page users
time_spent_old = df[df['landing_page'] == "old"]['time_spent_on_the_page'
```

In [43]:
```
print('The sample standard deviation of the time spent on the new page is
print('The sample standard deviation of the time spent on the new page is
```

```
The sample standard deviation of the time spent on the new page is: 1.82
The sample standard deviation of the time spent on the new page is: 2.58
```

Based on the sample standard deviations of the two groups, decide whether the population standard deviations can be assumed to be equal or unequal.

Step 5: Calculate the p-value

In [44]:
```
# complete the code to import the required function
from scipy.stats import ttest_ind

# write the code to calculate the p-value
test_stat, p_value =  ttest_ind(time_spent_new, time_spent_old, equal_var

print('The p-value is', p_value)
```

```
The p-value is 0.0001392381225166549
```

Step 6: Compare the p-value with α

In [45]:
```
# print the conclusion based on p-value
if p_value < 0.05:
    print(f'As the p-value {p_value} is less than the level of significan
else:
    print(f'As the p-value {p_value} is greater than the level of signifi
```

```
As the p-value 0.0001392381225166549 is less than the level of significan
ce, we reject the null hypothesis.
```

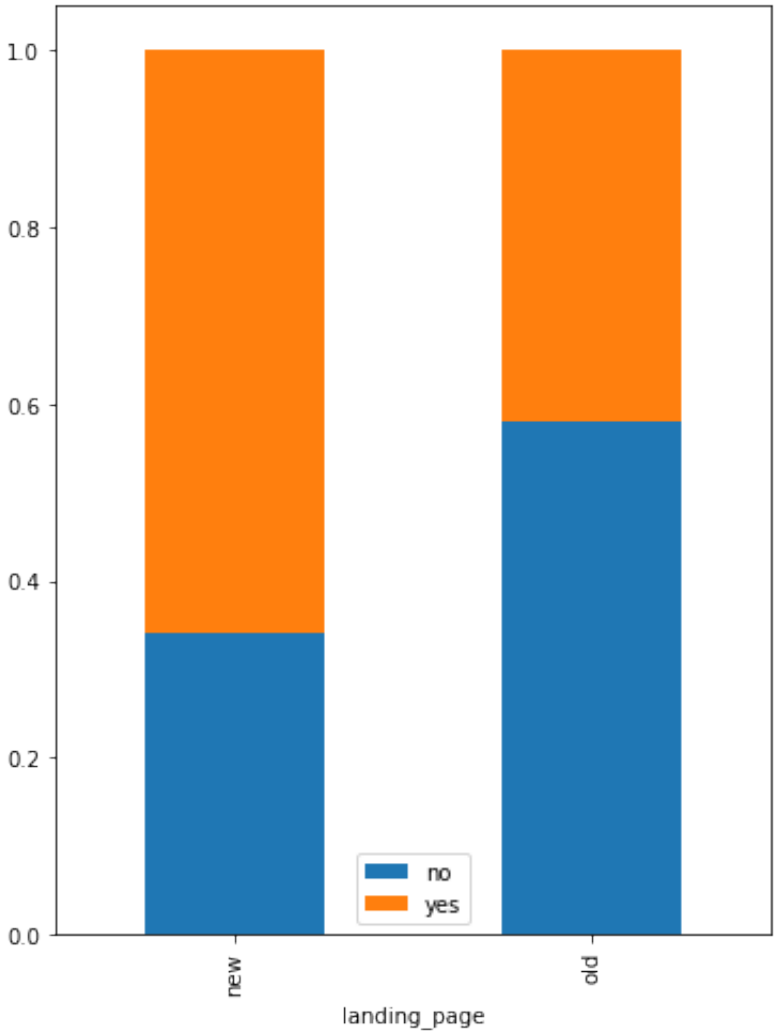
Step 7: Draw inference

Since the p-value is less than the significance, there is enough evidence to conclude that we reject the null hypothesis in favour of alternative hypothesis that the users spent more time on the new landing page

2. Is the conversion rate (the proportion of users who visit the landing page and get converted) for the new page greater than the conversion rate for the old page?

Perform Visual Analysis

In [48]:
```
# complete the code to visually compare the conversion rate for the new p
pd.crosstab(df["landing_page"],df["converted"],normalize='index').plot(ki
plt.legend()
plt.show()
```



Step 1: Define the null and alternate hypotheses

The null hypothesis for this situation would be that the conversion rate for the new page is equal to or less than the conversion rate for the old page.

The alternative hypothesis would be that the conversion rate for the new page is greater than the conversion rate for the old page.

$H_0 : p_{\text{new}} = p_{\text{old}}$

$H_a : p_{\text{new}} > p_{\text{old}}$

Step 2: Select Appropriate test

(This is a one-tailed test concerning two population proportions from two independent populations. **Based on this information, select the appropriate test**.)

The appropriate test for this situation would be a z-test for the difference in two proportions. This test is used to compare the proportion of success of two independent populations. Since this is a one-tailed test and the population is normal, using a z-test would be appropriate.

Step 3: Decide the significance level

As given in the problem statement, we select α = 0.05.

Step 4: Collect and prepare data

In [49]:
```
# calculate the number of converted users in the treatment group
new_converted = df[df['group'] == 'treatment']['converted'].value_counts(
# calculate the number of converted users in the control group
old_converted = df[df['group'] == 'control']['converted'].value_counts()[

n_control = df.group.value_counts()['control'] # total number of users in
n_treatment = df.group.value_counts()['treatment'] # total number of user

print('The numbers of users served the new and old pages are {0} and {1}
```

```
The numbers of users served the new and old pages are 50 and 50 respectiv
ely
```

Step 5: Calculate the p-value

In [53]:
```
# complete the code to import the required function
from statsmodels.stats.proportion import proportions_ztest

# write the code to calculate the p-value
test_stat, p_value = proportions_ztest([new_converted, old_converted] , [


print('The p-value is', p_value)
```

```
The p-value is 0.008026308204056278
```

Step 6: Compare the p-value with α

In [54]:
```
# print the conclusion based on p-value
if p_value < 0.05:
    print(f'As the p-value {p_value} is less than the level of significan
else:
    print(f'As the p-value {p_value} is greater than the level of signifi
```

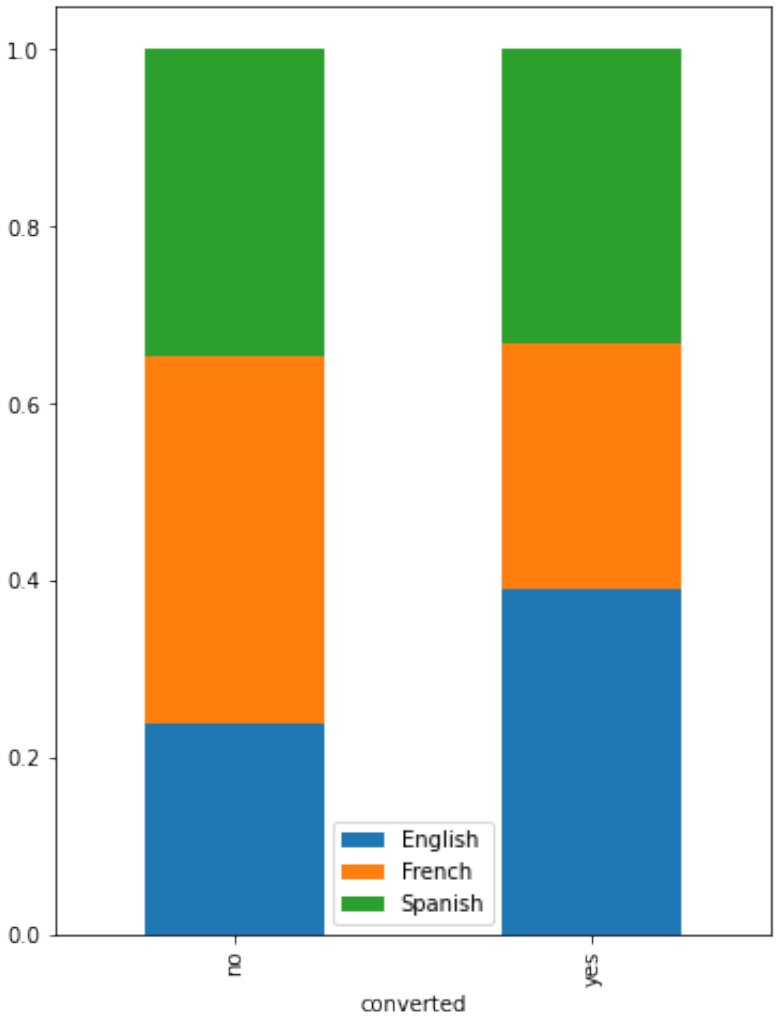
```
As the p-value 0.008026308204056278 is less than the level of significanc
e, we reject the null hypothesis.
```

Step 7: Draw inference

3. Does the converted status depend on the preferred language?

Perform Visual Analysis

In [57]:
```
# complete the code to visually plot the dependency between conversion st
pd.crosstab(df['converted'],df['language_preferred'],normalize='index').p
plt.legend()
plt.show()
```



Step 1: Define the null and alternate hypotheses

H_0 : The converted status is independent of the preferred language.

H_a : The converted status depends on the preferred language.

Step 2: Select Appropriate test

(This is a problem of the test of independence, concerning two categorical variables - converted status and preferred language. **Based on this information, select the appropriate test.**)

The appropriate test for this situation would be chi-squared test for independence. This test is used to determine if there is a significant association between two categorical variables. The chi-squared test compares the observed frequencies of the two variables in a contingency table to the expected frequencies if the two variables were independent. Since you are trying to determine if there is a relationship between the converted status and the preferred language, the chi-squared test for independence would be the appropriate test to use.

Step 3: Decide the significance level

As given in the problem statement, we select α = 0.05.

Step 4: Collect and prepare data

In [58]:
```
# complete the code to create a contingency table showing the distributio
contingency_table = pd.crosstab(df["language_preferred"], df["converted"]

contingency_table
```

Out[58]:

language_preferred	converted	no	yes
English		11	21
French		19	15
Spanish		16	18

Step 5: Calculate the p-value

In [60]:
```
# complete the code to import the required function
from scipy.stats import chi2_contingency

# write the code to calculate the p-value
chi2, p_value, dof, exp_freq = chi2_contingency(contingency_table)   # #c

print('The p-value is', p_value)
```

```
The p-value is 0.2129888748754345
```

Step 6: Compare the p-value with α

In [61]:
```
# print the conclusion based on p-value
if p_value < 0.05:
    print(f'As the p-value {p_value} is less than the level of significan
else:
    print(f'As the p-value {p_value} is greater than the level of signifi
```

```
As the p-value 0.2129888748754345 is greater than the level of significan
ce, we fail to reject the null hypothesis.
```

Step 7: Draw inference

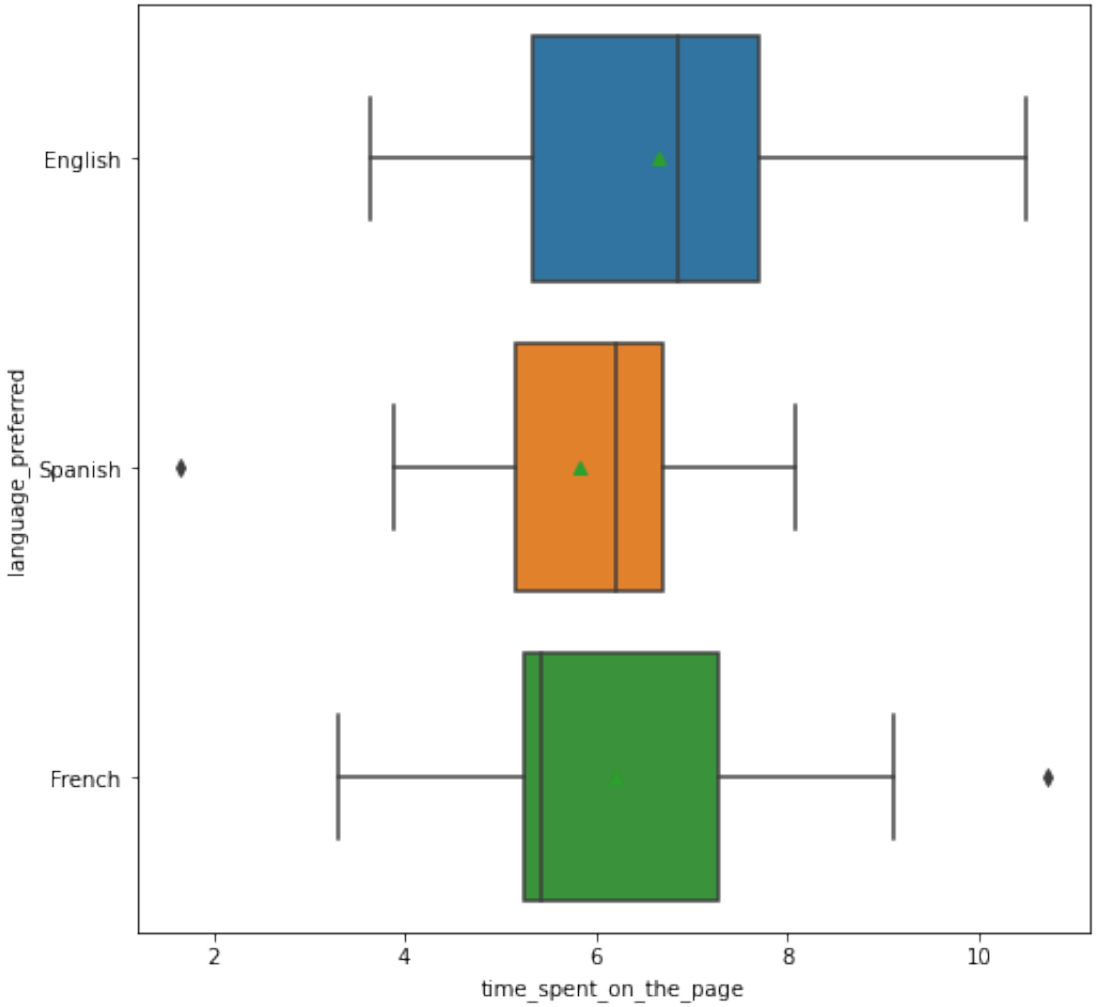
Since the p-value is greater than significance level, we fail to reject the null hypothesis. There is no enough evidence to support the alternative hypothesis that the conversion rate depends on users preferred language. hence, the language preference of the users and the conversion rate are independent.

4. Is the time spent on the new page same for the different language users?

Perform Visual Analysis

In [64]:
```
# create a new DataFrame for users who got served the new page
df_new = df[df['landing_page'] == 'new']
```

In [65]:
```
# complete the code to visually plot the time spent on the new page for d
plt.figure(figsize=(8,8))
sns.boxplot(x = "time_spent_on_the_page", y = "language_preferred", showm
plt.show()
```



In [66]:
```
# complete the code to calculate the mean time spent on the new page for
df_new.groupby(["language_preferred"])['time_spent_on_the_page'].mean()
```

Out[66]:
```
language_preferred
English    6.663750
French     6.196471
Spanish    5.835294
Name: time_spent_on_the_page, dtype: float64
```

Step 1: Define the null and alternate hypotheses

H_0 : The mean time spent by Spanish, French and English language users is equal.

H_a : At least one mean is unequal.

Step 2: Select Appropriate test

(This is a problem, concerning three population means. **Based on this information, select the appropriate test to compare the three population means.**)

The appropriate test for this situation would be ANOVA (Analysis of Variance) test, which is used to compare the means of three or more independent groups. This test can be used to determine if there is a significant difference in the mean time spent on the new page among users of different languages. The ANOVA test can also help to determine which group or groups are responsible for the observed difference, if any. It is a parametric test and it assumes that the data is normally distributed, independent and have equal variances among the groups.

Step 3: Decide the significance level

As given in the problem statement, we select α = 0.05.

Step 4: Collect and prepare data

In [72]:
```
# create a subsetted data frame of the time spent on the new page by Engl
time_spent_English = df_new[df_new['language_preferred']=="English"]['tim
# create subsetted data frames of the time spent on the new page by Frenc
time_spent_French = df_new[df_new['language_preferred']=="French"]['time_
time_spent_Spanish = df_new[df_new['language_preferred']=="Spanish"]['tim
```

Step 5: Calculate the p-value

In [69]:
```
# complete the code to import the required function
from scipy.stats import levene

# write the code to calculate the p-value
test_stat, p_value = levene(time_spent_English, time_spent_French, time_s

print('The p-value is', p_value)
```

```
The p-value is 0.46711357711340173
```

Step 6: Compare the p-value with α

In [70]:
```
# print the conclusion based on p-value
if p_value < 0.05:
    print(f'As the p-value {p_value} is less than the level of significan
else:
    print(f'As the p-value {p_value} is greater than the level of signifi
```

```
As the p-value 0.46711357711340173 is greater than the level of significa
nce, we fail to reject the null hypothesis.
```

Step 7: Draw inference

As the p-value is greater than the level of significance, we cannot reject the null hypothesis. Hence, there is enough evidence to support the claim that the mean time spent on the new page is equal across the different language users.

Conclusion and Business Recommendations
