1. The problem asks to test if the distribution has a mean value of 7725 kJ, and so we should test if the sample mean is equal or not equal to 7725 kJ, therefore a two-tailed test is appropriate here. From my calculation, the p-value is 0.018, which is smaller than the significant level 0.05, so we can reject the null hypothesis. Thus, H0 is rejected, supporting that the women's energy intake deviates systematically from a recommended value of 7725 kJ.

Q1:

Null Hypothesis: the women's energy intake does not deviate systematically from a recommended value of 7725 kJ; H0: population mean=7725

Alternative Hypothesis: the women's energy intake deviates systematically from a recommended value of 7725 kJ. H1: population mean≠7725

```
energy_intake=[5260,5470,5640,6180,6390,6515,6805,7515, 7515, 8230, 8770]
alpha=0.05
=7725
n=len(energy_intake)
samp_mean=sum(energy_intake)/n
var=sum((x - samp_mean) ** 2 for x in energy_intake)/(n-1)
samp_std=math.sqrt(var)
SEM=sts.sem(energy_intake)
print('Sample mean: '+ str(samp_mean)+'\nSample standard deviation: '+str(samp_std)+'\nSEM: '+str(SEM))
```

```
Sample mean: 6753.636363636364
Sample standard deviation: 1142.1232221373727
SEM: 344.3631083801271
```

```
DoF=n-1
print('Degrees of freedom: '+str(DoF))
t_stats=stats.ttest_1samp(energy_intake, popmean=recom_val, alternative='two-sided')
print('t statistic is: '+str(t_stats[0]),'\np-value is: '+str(t_stats[1]))
```

```
Degrees of freedom: 10
t statistic is: -2.8207540608310198
p-value is: 0.018137235176105812
```

2. The conclusion is that Guinness in an Irish pub has a significantly better taste than elsewhere. This is supported by conducting a two-sample t-test, and a significant p-value< 0.01 is obtained (shown below). Since these two samples are independent (not from the same subject), we used a two-sample independent t-test. Plus, the question asks whether Ireland is greater than elsewhere, and this tells us a right-tailed test is required.

Q2: Two-sample right-tailed t-test

Null hypothesis: Guinness served in an Irish pub tastes sam as pints served elsewhere. Mean Ireland = Mean Elsewhere

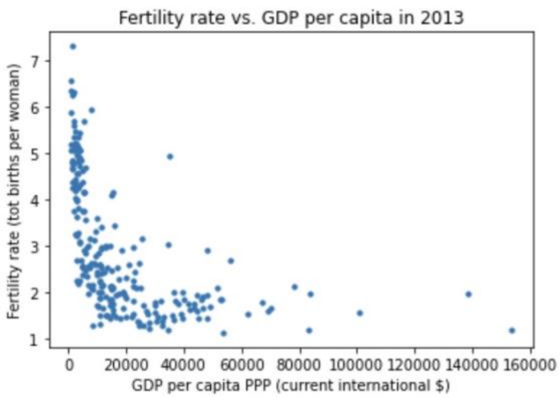
Alternative hypothesis: Guinness in an Irish pub tastes significantly better than pints served elsewhere. Mean Ireland - Mean Elsewhere >0

Set up the mean, sd, and sample size of Ireland as m1,sd1,n1. Elsewhere: m2, sd2, n2

```
n1=42;n2=61
m1=74;m2=57
sd1=7.4;sd2=7.1
df=n1+n2-2
## method 1
t=(m1-m2)/math.sqrt((sd1**2/n1)+(sd2**2/n2))
p_val=scipy.stats.t.sf(abs(t), df)
print('method 1 calulcate manually: '+'t-statistic: '+str(t)+'\np-value: '+str(p_val))
## method 2
t_stats=scipy.stats.ttest_ind_from_stats(mean1=m1, std1=sd1, nobs1=n1,
                        mean2=m2, std2=sd2, nobs2=n2,alternative='greater')
print('method 2 use scipy library: '+'t-statistic: '+str(t_stats.statistic)+'\np-value: '+str(t_stats.pvalue))
```

```
method 1 calulcate manually: t-statistic: 11.647653131319812
p-value: 1.095517028264447e-20
method 2 use scipy library: t-statistic: 11.73775770205081
p-value: 6.979768077580737e-21
```

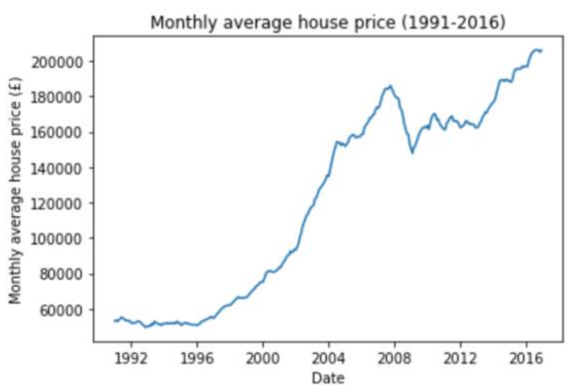
3. The correlation coefficient is small around -0.279. The scatter plot also showed a weak negative correlation between fertility rate and GDP per capita. In a nutshell, these show that fertility rate and GDP per capita are slightly negatively correlated in 2013. However, this only tests out in 2013, this might not be true if more years are included.

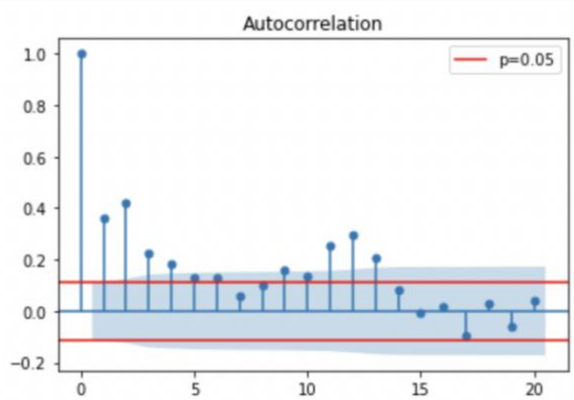



```python
corr_coeff=GDP_PPP['2013'].corr(Fertility['2013'])
print('The correlation coefficient is '+str(f'{corr_coeff:.3}'))
```

```
The correlation coefficient is -0.279
```

4. From the time series plot, we can see that the monthly average price rises drastically from 1996 to 2008, drops suddenly in 2008, and rises after 2010.





The ACF plot of house monthly price shows the seasonality as the autocorrelation drops quickly in the first 6 lags and then rises from the 7th to 12th lags, and then drops. Thus, both plots show the seasonality of house prices. From my calculation, the annualized house return is 5.354%.

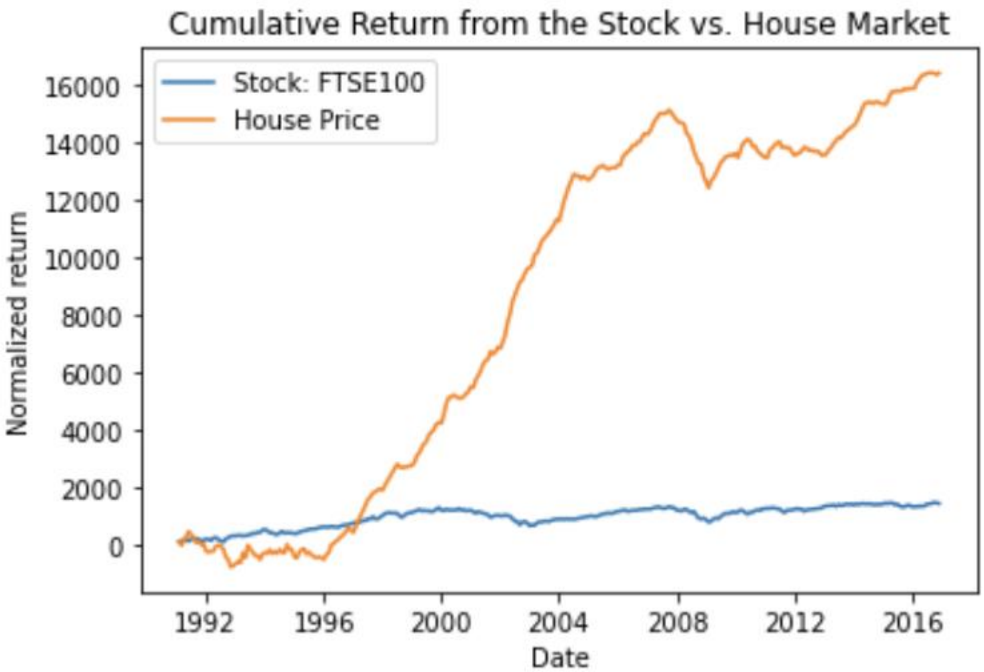
```python
## Annualized FSTE return
tt=np.arange(1,312)
ann_temp = lambda t:1+return_FTSE_lst[t]
ann_lst=np.array(list(map(ann_temp,tt)))
annual_return_FTSE=(np.prod(ann_lst)**(12/312)-1)
# annual_return_FTSE=((FTSE100_cums[-1]/FTSE100_cums[0])**(1/(312/12))-1)*100
print(f'The annualized FTSE return is {annual_return_FTSE:.3%}')
```

```
The annualized FTSE return is 4.463%
```

```python
## Annualized house return
ann_temp = lambda t:1+return_house_lst[t]
ann_lst=np.array(list(map(ann_temp,tt)))
annual_return_house=(np.prod(ann_lst)**(12/312)-1)
print(f'The annualized house return is {annual_return_house:.3%}')
```

```
The annualized house return is 5.354%
```

5. The average annualized FSTE return is 4.463%. House has around 1% return greater than stock. From the scatter plot and the calculated annualized return, I think it is better to invest in the UK house rather than the stock market in a long term like the time period here across 26 years. However, the stock price is more stable compared to the house market. Thus, if the investor wants to get a stable return, I would suggest investing in the stock.




```
## Annualized FSTE return
tt=np.arange(1,312)
ann_temp = lambda t:1+return_FTSE_lst[t]
ann_lst=np.array(list(map(ann_temp,tt)))
annual_return_FTSE=(np.prod(ann_lst)**(12/312)-1)
# annual_return_FTSE=((FTSE100_cums[-1]/FTSE100_cums[0])**(1/(312/12))-1)*100
print(f'The annualized FTSE return is {annual_return_FTSE:.3%}')
```

```
The annualized FTSE return is 4.463%
```

```
## Annualized house return
ann_temp = lambda t:1+return_house_lst[t]
ann_lst=np.array(list(map(ann_temp,tt)))
annual_return_house=(np.prod(ann_lst)**(12/312)-1)
print(f'The annualized house return is {annual_return_house:.3%}')
```

```
The annualized house return is 5.354%
```