

Uswitch Energy Comparison

Choosing an energy supplier is a difficult job. There are many suppliers with different plans and prices that vary by how much energy a customer consumes.

Your task is to help make the customer's decision easier by writing a program that:

- Prices plans on the market according to how much energy is used
- Calculates how much energy is used according to how much a customer spends each month on a specific plan

We provide automated acceptance scenarios for the expected program behaviour, available for Ruby and JavaScript. Please choose the language you feel most comfortable with.

This document outlines how to execute the scenarios in order to verify correct solution behaviour. You may find it helpful to write additional unit tests. We also include a little background information on the problem domain.

The plans data

We provide some example energy market data in `plans.json`

Below is an example of one such plan;

```
[
  {
    "supplier": "sse",
    "plan": "standard",
    "rates": [
      {"price": 13.5, "threshold": 150},
      {"price": 11.1, "threshold": 100},
      {"price": 10}
    ],
    "standing_charge": 9
  }
  ...
]
```

Plans contain a supplier & plan, which are identifier of who supplies that energy and name of the plan. They also include rates which indicate how much the customer will be charged for each kilowatt-hour (kWh) of energy that they use. Additionally, plans may also include a daily standing charge.

A plan can have multiple rates which charge a different price at each rate. Each rate also includes a `threshold` which is the amount of kWhs a user can consume at that price before moving to the next rate. The last rate will not have a threshold. Rates without a threshold have no limit on kWh usage.

In the example above, the first 150kWh will be charged at 13.5p/kWh, the next 100kWh will be charged at 11.1p/kWh and all subsequent consumption will be charged at 10p/kWh.

Plans can also (sometimes) include a `standing_charge`, which is a static amount in pence which is added as a daily cost. For example, a standing charge of 10p/day would cost £36.50 over the year.

Note that:

- Prices are stated in pence **exclusive** of VAT.
- Standing charge is a daily charge stated in pence exclusive of VAT and is applied regardless of consumption.

- VAT for Energy is rated at 5%.

Your solution

Your task is to produce a program that implements two commands

- price - To return the cost of each plan for a given kWh usage, sorted by cheapest
- usage - To return the amount of kWh you can use for a given plan and monthly spend

We have provided a skeleton of the code to get you started in both JS & Ruby. This is available in the `/js/lib/energy-market.js` & `/ruby/lib/energy_market.rb` directories.

You should update those files with your solution.

Feature scenarios

To help validate your program works, we have outlined the intended behaviour for both in **feature files**. **Feature files**, a natural language description of the intended behaviour of your program given different inputs.

Feature: Price plans based on usage

Energy switchers want to see the cheapest energy tariffs based on their usage

Scenario: Find the cheapest energy plans when usage low

Given the plans provided

When annual usage is 1000 kwh

Then the cheapest plans are

provider	plan type	price
eon	variable	108.68
edf	fixed	111.25
ovo	standard	120.23
bg	standing-charge	121.33

Scenario: Find the cheapest energy plans when usage high

Given the plans provided

When annual usage is 2000 kwh

Then the cheapest plans are

provider	plan type	price
edf	fixed	205.75
eon	variable	213.68
bg	standing-charge	215.83
ovo	standard	235.73

Feature: Calculate annual usage in kWh from a monthly spend

Scenario: Standard Plan

Given the plans provided

When supplier name is "ovo"

And plan type is "standard"

And monthly spend is 1000 pounds

Then annual usage is 103855 kWh

Scenario: Fixed Plan

Given the plans provided

When supplier name is "edf"

And plan type is "fixed"

```
And monthly spend is 350 pounds
Then annual usage is 44267 kWh
```

```
Scenario: Standing Charge
  Given the plans provided
  When supplier name is "bg"
  And plan type is "standing-charge"
  And monthly spend is 120 pounds
  Then annual usage is 14954 kWh
```

Checking your solution

You can verify that your program solution works by executing the automated scenarios

Ruby You will require bundler

You can test your installation with: `bundle -v`

To run the feature scenarios;

```
cd {coding-test-dir}/ruby
bundle
cucumber
```

JS You will require a working node and npm installation. Verify that Node.js is installed properly:

```
node -v
npm -v
```

Execute the scenarios

```
cd {coding-test-dir}/js
npm install
npm test
```

Submitting

Your code should be submitted in a tarball or zip that contains your full project and emailed to our recruiting team.

We will verify your solution passess the automated scenarios provided.

Tips

You are welcome to write your solution in a language you feel confident in. Your program must pass the automated scenarios provided and be well written.

During the in-office interviews we will pair with you to extend the problem (and your program). You should re-read, and feel comfortable with your code before you come in if you've not looked at it for a while.

Please take your time to solve the problem fully, we appreciate people have day jobs and other commitments. Please let us know if you need more time, or if something is unclear.

Please do not publish your solution, for example on your blog or source control site.