

```
In [29]: 1 # # Get list of index values for all invalid State code rows
2 # invalid_1 = list(has_address[has_address['State'].isin(invalid_states)].index)
```

4. Check nulls

Answer: The resulting table shows the rows in `charges` where the `'State'` code OR `'Country'` code is **NULL**.

```
In [30]: 1 # Find all rows where State code OR Country code is NULL
2 has_address[(has_address['State'].isnull()) | (has_address['Country'].isnull())]
```

Out[30]:

	Address 1	Address 2	City	State	Zip	Country
218	173 Main St.	NaN	Alberta	NaN	NaN	CA
1720	731 Main St.	Seuzach Switzerland	NaN	NaN	NaN	NaN
1758	234 Main St.	NaN	NaN	NaN	NaN	NaN
1812	992 Main St.	NaN	Vancouver	BC	CA	NaN
2175	127 Main St.	NaN	NaN	NaN	NaN	NaN

```
In [31]: 1 # Get list of index values for all null rows
2 invalid_2 = list(has_address[(has_address['State'].isnull()) | (has_address['Country'].isnull())].index)
```

5. Final overview

All invalid rows were in one of 2 groups:

1. Invalid `'Country'` code - either **null** or **USA** which is NOT the correct country code according to the `countries` table.
2. Invalid `'State'` code - either **null** or an **incorrect code**: `['XY', 'VW', 'CD', 'AE', 'UX', 'AW']`.

6. Other observations/EDA

- Notice `has_address.iloc[1720]` which has city/state/country information in the `Address 2` column instead of in the *appropriate* columns. Based on the `countries` table, the country code for Switzerland would be `CH`.