Setup

To install everything, open your terminal and run: `pip install altair jupyterlab altair_data_server`

In [1]:
```
import pandas as pd
import altair as alt
from altair import datum

# Makes nice SVG images
# alt.renderers.enable('jupyterlab', embed_options={'renderer': 'svg'})

# Avoids writing all the data to the notebook or disk.
# See https://altair-viz.github.io/user_guide/faq.html#local-data-server
# Note that this may not work on some cloud-based Jupyter notebook services.
alt.data_transformers.enable('data_server')
```

Out[1]:
```
DataTransformerRegistry.enable('data_server')
```

Background

The census tract for the South End begins just on the SE side of Columbus Avenue, as shown in this map:

In [2]:
```
df = pd.read_csv('south_end.csv')
```

In [3]:
```
df.columns
```

Out[3]:
```
Index(['Category', 'Subcategory', 'Decade', 'Count', 'Percent'], dtype='object')
```

In [4]:
```
df[['Category','Subcategory']].drop_duplicates()
```

Out[4]:

	Category	Subcategory
0	Population	NaN
1	Age	0-9 years
2	Age	10-19 years
3	Age	20-34 years
4	Age	35-54 years
5	Age	55-64 years
6	Age	65 years and over
7	Educational Attainment (age 25+)	less than High School
8	Educational Attainment (age 25+)	High School or GED
9	Educational Attainment (age 25+)	Some College or Associate's Degree
10	Educational Attainment (age 25+)	Bachelor's Degree or Higher
11	Nativity	Foreign Born
12	Race/ Ethnicity	White
13	Race/ Ethnicity	Black/ African American
14	Race/ Ethnicity	Hispanic
15	Race/ Ethnicity	Asian/PI
16	Race/ Ethnicity	Other
17	Labor Force (age 16+)	Male
18	Labor Force (age 16+)	Female
19	Housing Tenure	Occupied Housing Units
20	Housing Tenure	Owner-occupied
21	Housing Tenure	Renter-occupied

In [5]:
```
df
```

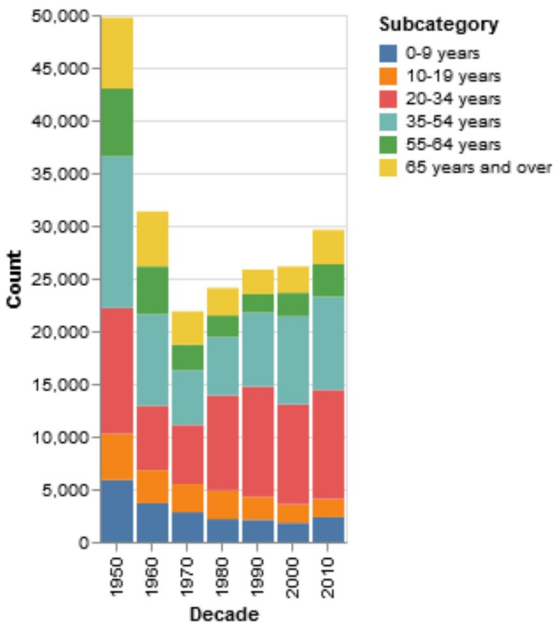
Out[5]:

	Category	Subcategory	Decade	Count	Percent
0	Population	NaN	1950	49753	NaN
1	Age	0-9 years	1950	5870	0.12
2	Age	10-19 years	1950	4387	0.09
3	Age	20-34 years	1950	11947	0.24
4	Age	35-54 years	1950	14374	0.29
...	...	...	...	...	...
149	Labor Force (age 16+)	Male	2010	-	NaN
150	Labor Force (age 16+)	Female	2010	-	NaN
151	Housing Tenure	Occupied Housing Units	2010	15629	NaN
152	Housing Tenure	Owner-occupied	2010	5702	0.36
153	Housing Tenure	Renter-occupied	2010	9927	0.64

154 rows × 5 columns

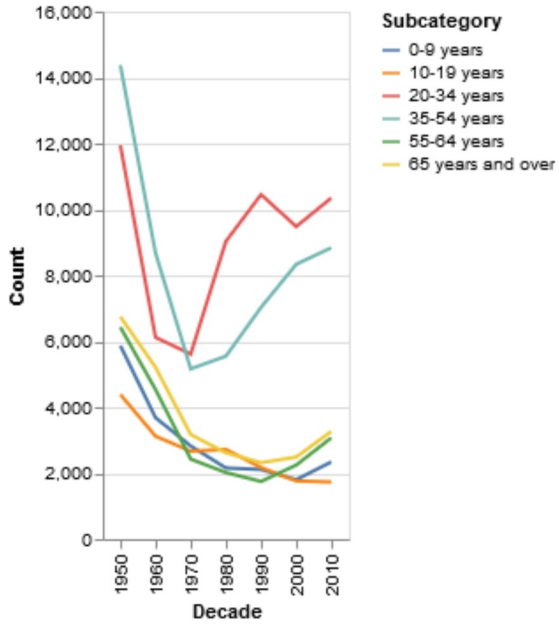
In [6]:
```
alt.Chart(df).mark_bar().encode(
    x='Decade:O',
    y='Count:Q',
    color='Subcategory:N',
    order=alt.Order(
      'Subcategory:N',
      sort='ascending'
    )
).transform_filter(
    (datum.Category == 'Age')
)
```

Out[6]:



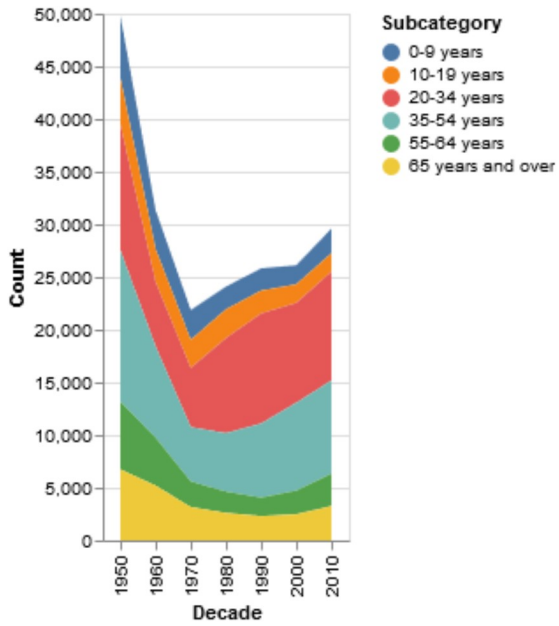
In [7]:
```
alt.Chart(df).mark_line().encode(
    x='Decade:O',
    y='Count:Q',
    color='Subcategory'
).transform_filter(
    (datum.Category == 'Age')
)
```

Out[7]:



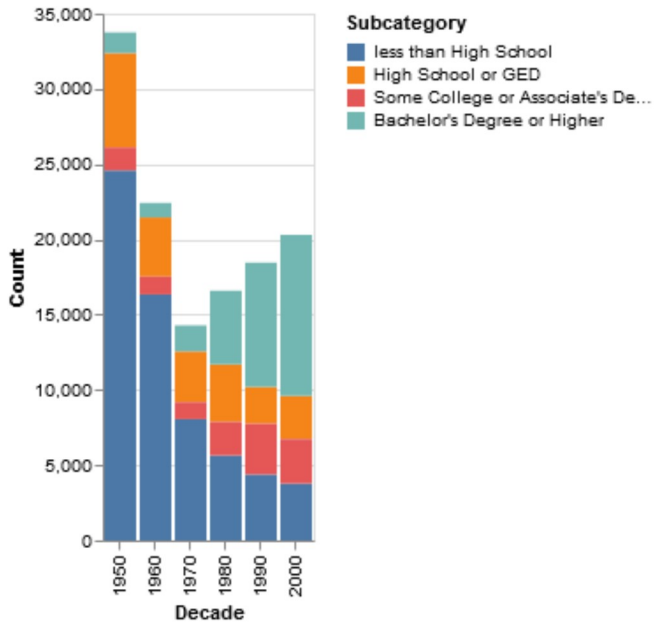
In [8]:
```
alt.Chart(df).mark_area().encode(
    x='Decade:O',
    y='Count:Q',
    color='Subcategory'
).transform_filter(
    (datum.Category == 'Age')
)
```

Out[8]:



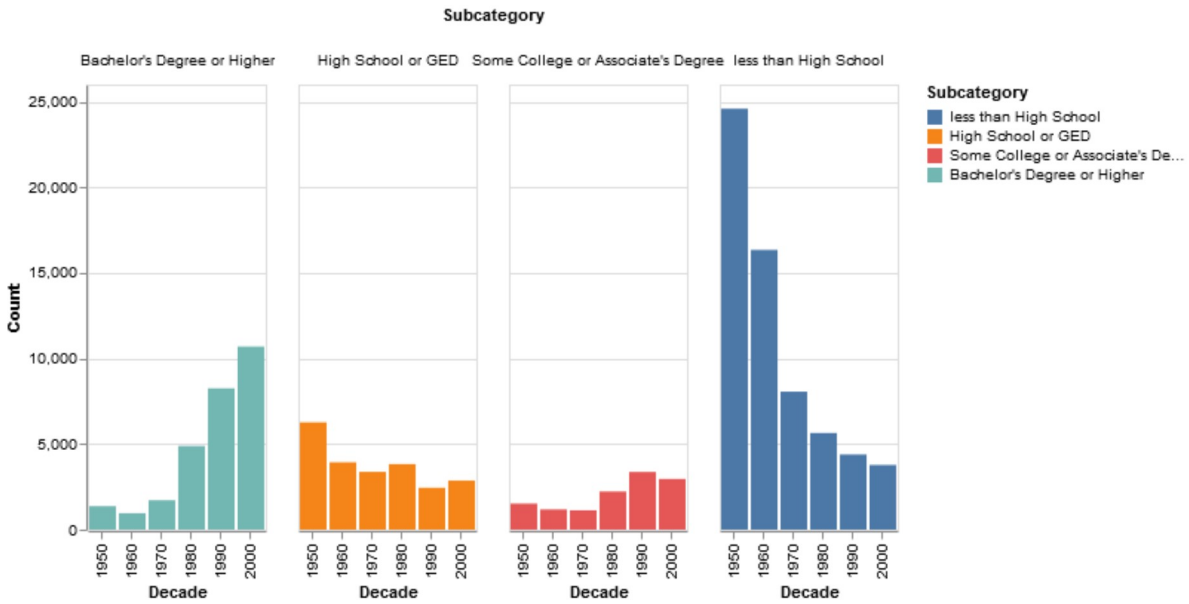
In [9]:
```
alt.Chart(df).mark_bar().encode(
    x='Decade:O',
    y='Count:Q',
    color=alt.Color('Subcategory', sort=['less than High School', 'High School or GED', "Some College or Associate's Degree", "Bachelor's Degree or Higher"])
).transform_filter(
    (datum.Category == 'Educational Attainment (age 25+)')
)
```

Out[9]:



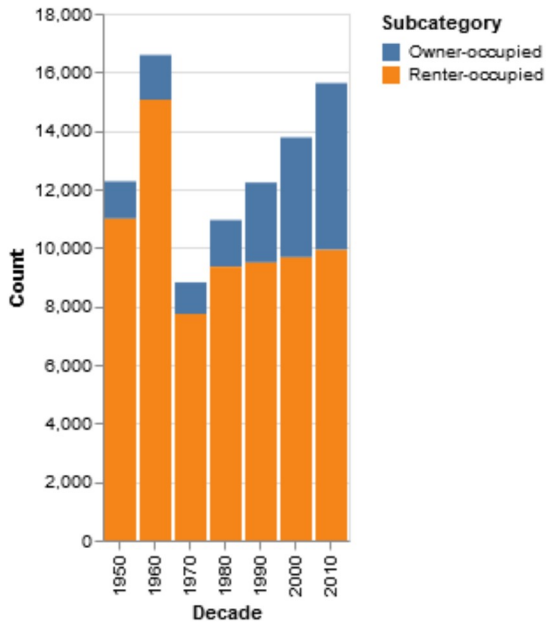
In [10]:
```
alt.Chart(df).mark_bar().encode(
    x='Decade:O',
    y='Count:Q',
    color=alt.Color('Subcategory:N', sort=['less than High School', 'High School or GED', "Some College or Associate's Degree", "Bachelor's Degree or Higher"]),
    column='Subcategory:N'
).transform_filter(
    (datum.Category == 'Educational Attainment (age 25+)')
)
```

Out[10]:



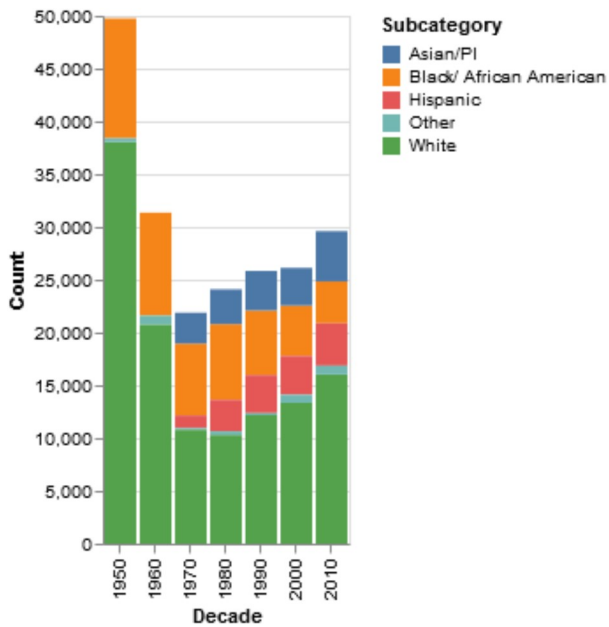
In [11]:
```
alt.Chart(df).mark_bar().encode(
    x='Decade:O',
    y='Count:Q',
    color='Subcategory'
).transform_filter(
    (datum.Category == 'Housing Tenure') & (datum.Subcategory != 'Occupied Housing Units')
)
```

Out[11]:



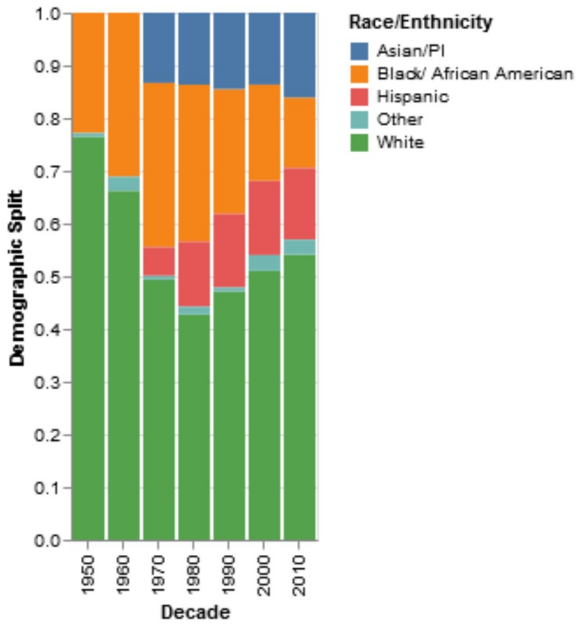
In [12]:
```
alt.Chart(df).mark_bar().encode(
    x='Decade:O',
    y='Count:Q',
    color='Subcategory'
).transform_filter(
    (datum.Category == 'Race/ Ethnicity')
)
```

Out[12]:



In [13]:
```
chart_race = alt.Chart(df).mark_bar().encode(
    x='Decade:O',
    y=alt.Y('Count:Q', stack='normalize', title='Demographic Split'),
    color=alt.Color('Subcategory', title='Race/Enthnicity'),
    tooltip=['Subcategory', 'Count', 'Decade']
).transform_filter(
    (datum.Category == 'Race/ Ethnicity')
)
chart_race
```

Out[13]:



In [14]:
```
chart_race.save('chart_race.html', embed_options={'renderer':'svg'})
```