Import required package

```
from matplotlib.pyplot import figure
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
```

Global Settings

```
figure(num=None, figsize=(9, 10), dpi=80, facecolor='w', edgecolor='k')
```

Define stack bar chart function

```
def stacked_bar_chart_hospital_death(data_source, x_axis, show_counts=False,
                                     legend_position="upper left",
                                     rotate_x=None, x_axis_fontsize=10,
                                     bar_width=0.6):
    live = np.array(data_source["Live"])
    death = np.array(data_source["Death"])
    bar1 = plt.bar(data_source[x_axis].unique().tolist(), death,
                   width=bar_width, label='Death', bottom=live)
    bar2 = plt.bar(data_source[x_axis].unique().tolist(), live,
                   width=bar_width, label='Live')
    plt.xlabel(x_axis)
    plt.xticks(fontsize=x_axis_fontsize)
    plt.ylabel("% of Patient")
    plt.legend(loc=legend_position)
    plt.title("Hospital Death By {0}".format(x_axis), fontsize=16)

    if rotate_x is not None:
        plt.setp(plt.gca().get_xticklabels(), rotation=rotate_x,
                 horizontalalignment='right')

    if show_counts:
        print(bar1, bar2)
        for r1, r2 in zip(bar1, bar2):
            x1, h1, w1 = r1.get_x(), r1.get_height(), r1.get_width()
            x2, h2, w2 = r2.get_x(), r2.get_height(), r2.get_width()
            plt.text(x2 + w2 / 2., h2 / 2., "%d" % h2, ha="center",
                     va="center", color="white", fontsize=12)
            plt.text(x1 + w1 / 2., (2.5 * h1 / 4) + h2, "%d" % h1,
                     ha="center", va="top", color="white", fontsize=12)

    plt.savefig("../images/{0}.png".format("death_status_by_{0}".
                format(x_axis.replace(" ", "_").lower())))

    plt.show()
```

Define histogram chart function

```
def histogram_chart(x_axis, name, bins=10):
    plt.hist(x_axis, bins=bins)
    plt.xlabel(name.title())
    plt.xticks(fontsize=10)
    plt.ylabel("No.Patient")
    plt.legend(loc="upper right")
    plt.title("Histogram of {0}".format(name.title()), fontsize=16)
    name = name.replace(" ", "_").lower()
    plt.savefig("images/{0}.png".format("histogram_by_{0}".format(name)))
    plt.show()
```

Define heatmap chart function

```
def plot_heatmap(data):
    sns.set(style="white")
    corr = data.corr()
    mask = np.triu(np.ones_like(corr, dtype=np.bool))
    # f, ax = plt.subplots(figsize=(15, 15))
    cmap = sns.diverging_palette(220, 10, as_cmap=True)
    sns.heatmap(corr, mask=mask, cmap=cmap, vmax=0.5, center=0,
                square=True, linewidths=.5, cbar_kws={"shrink": .5},
                annot=True)
    plt.show()
```