8.1

$$\frac{d[P]}{dt} = k_3[ES]$$

$$\frac{d[S]}{dt} = -k_1[E][S] + k_2[ES]$$

$$\frac{d[ES]}{dt} = -k_3[ES] - k_2[ES] + k_1[E][S]$$

$$\frac{d[E]}{dt} = -k_1[E][S] + k_2[ES] + k_3[ES]$$

8.2

When

$$k_1 = 100, k_2 = 600, k_3 = 150$$
$$E(0) = 1, S(0) = 10$$

Solve the equations above.

For applying fourth-order Runge Kutta method, firstly convert equations into standard form.

$$[ES] + [E] = E(0)$$
$$[ES] + [S] + [P] = S(0)$$

$$\frac{d[E]}{dt} = -k_1[E](S(0) - (E(0) - [E]) - [P]) + k_2(E(0) - [E]) + k_3(E(0) - [E])$$

$$\frac{d[E]}{dt} = -k_1[E](9 + [E] - [P]) + k_2(1 - [E]) + k_3(1 - [E])$$

The method of expressing [P] by [E] did not occur to me, so I could not proceed. But the code has been written as follows.

The Matlab code for the RK4 function is listed below.

```
function y = RK4(fun, x, y0)

y = 0 * x;
y(1) = y0;
h = x(2) - x(1);
n = length(x);
for m = 1 : n-1
k1 = fun(x(m), y(m));
k2 = fun(x(m)+h/2, y(m)+h*k1/2);
k3 = fun(x(m)+h/2, y(m)+h*k2/2);
k4 = fun(x(m)+h, y(m)+h*k3);
y(m+1) = y(m) + h*(k1 + 2*k2 + 2*k3 + k4) / 6;
end
End
```

The code for the solution is as follows

```
fun = @(x, y)(#Function#)
```

```
x = ;
y0 = ;
```

The code for the solution is as follows

```
y = RK4(fun, x, y0);

fig = gcf;
fig.Color = 'w';
fig.Position = [250, 250, 960, 540];

p = plot(x, y);
p.LineStyle = 'none';
p.Marker = 'p';
p.MarkerEdgeColor = 'r';
p.MarkerFaceColor = 'b';
p.MarkerSize = 8;

hold on, grid on
syms y(x)
equ = 2 * diff(y, x) == exp(-x) - y;
cond = y(0) == 1/2;
y = dsolve(equ, cond);

fplot(y, [0, 2])

xlabel('x', 'fontsize', 12);
ylabel('y', 'fontsize', 12);
title('RK4 for ODE', 'fontsize', 14);
legend({'Numerical solutions', 'Symbol Solution'}, 'fontsize', 12);
```