$$y' = \frac{2-2ty}{t^2+1}, \quad 0 \le t \le 1, \quad y(0)=1, \quad h=0.1$$

Taylor 2

$$f'(t, y(t))$$

$$= \frac{-2t^3y'+2t^2y-2ty'-4t-2y}{(t^2+1)^2}$$

$$= \frac{y'(-2t^3-2t)+2t^2y-4t-2y}{(t^2+1)^2}$$

$$= y' \cdot \frac{(-2t^3-2t)}{(t^2+1)^2} + \frac{2t^2 \cdot y-4t-2y}{(t^2+1)^2}$$

$$= \frac{-8t+6t^2y-2y}{(t^2+1)^2}$$

```
>> taylor_two
i        t                w
1        0.000000000      1.000000000
2        0.100000000      1.190000000
3        0.200000000      1.349218704
4        0.300000000      1.471259548
5        0.400000000      1.554619056
6        0.500000000      1.601919392
7        0.600000000      1.618402770
8        0.700000000      1.610385572
9        0.800000000      1.584099540
10       0.900000000      1.545025327
11       1.000000000      1.497628914
```

```
%%Section5.3 #9(a)
%% Inputs

a = 0;             % left endpoint
b = 1;             % right endpoint
h = 0.1;           % stepsize
N = (b-a)/h;       % the number of steps
alpha = 1;         % initial y value
f = @(t,y) (2-2*t*y)/(t^2+1);         % as in dy/dt = f(t,y);
df = @(t,y) (-8*t+6*t^2*y-2*y)/(t^2+1)^2;

%% Order 2

t = zeros(1,N+1);        % stores all the t values
w = zeros(1,N+1);        % stores all the approximation values for order 2

t(1) = a;
w(1) = alpha;
fprintf('i\t t\t\t\t w\t \n')

for i=1:N+1
    w(i+1) = w(i) + h*f(t(i),w(i)) + (h^2/2)*df(t(i),w(i));
    t(i+1) = a + i*h;
    fprintf('%d\t%.9f\t%.9f\n',i,t(i),w(i))

end
```