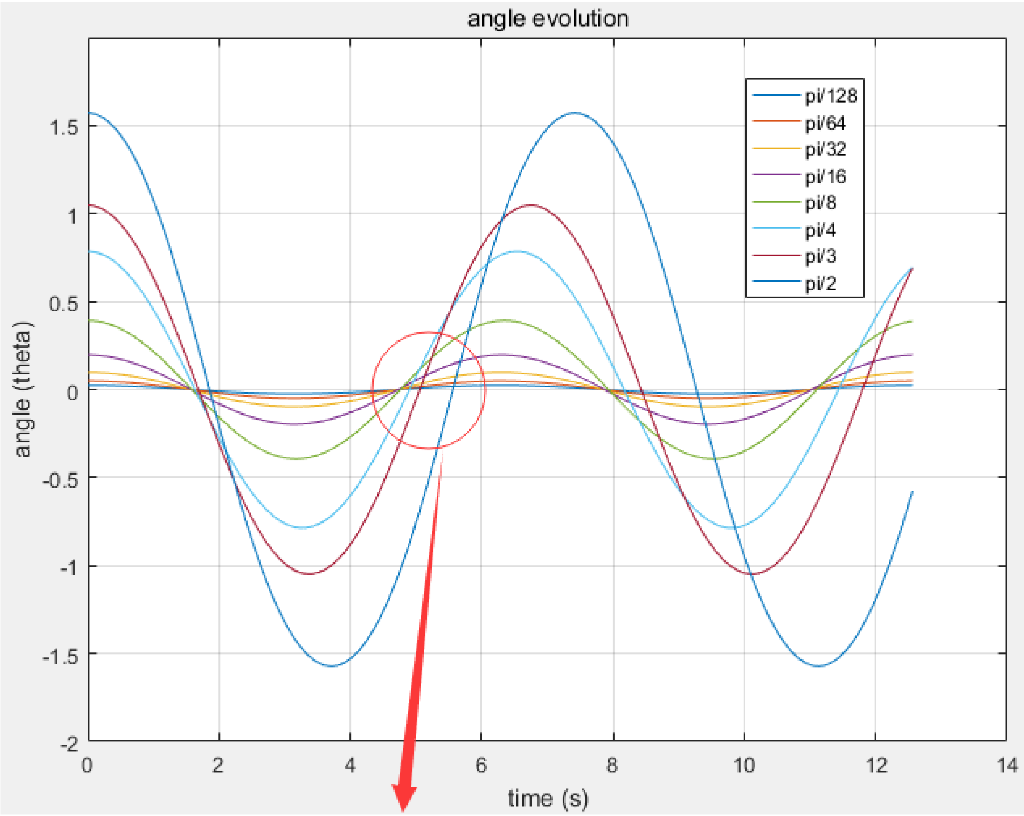
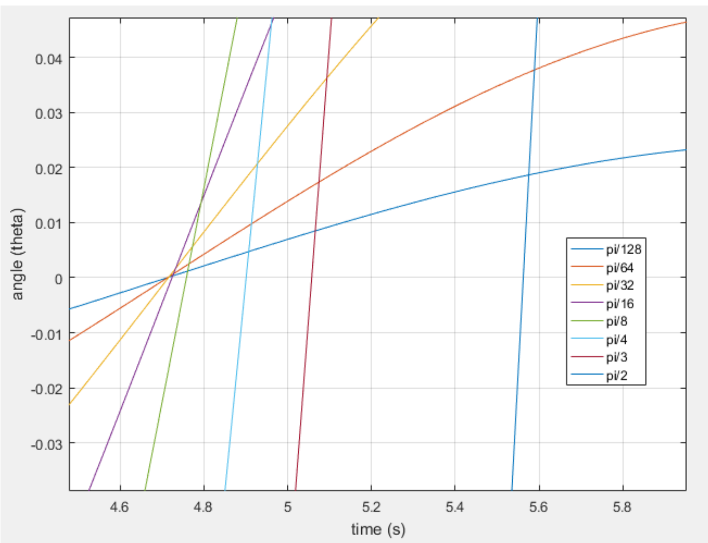
2.a) denote the angle θ by y_1, $\frac{d\theta}{dt}$ by y_2, we can write the equation of motion as

$$\frac{d}{dt}\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} y_2 \\ \frac{g}{l}\sin(y_1) \end{pmatrix}$$

For simplicity, we just set g/l = 1. Applying the rk4 method, we calculate the y_1, y_2 at different time t, respect to different initial angles.(step:0.01)

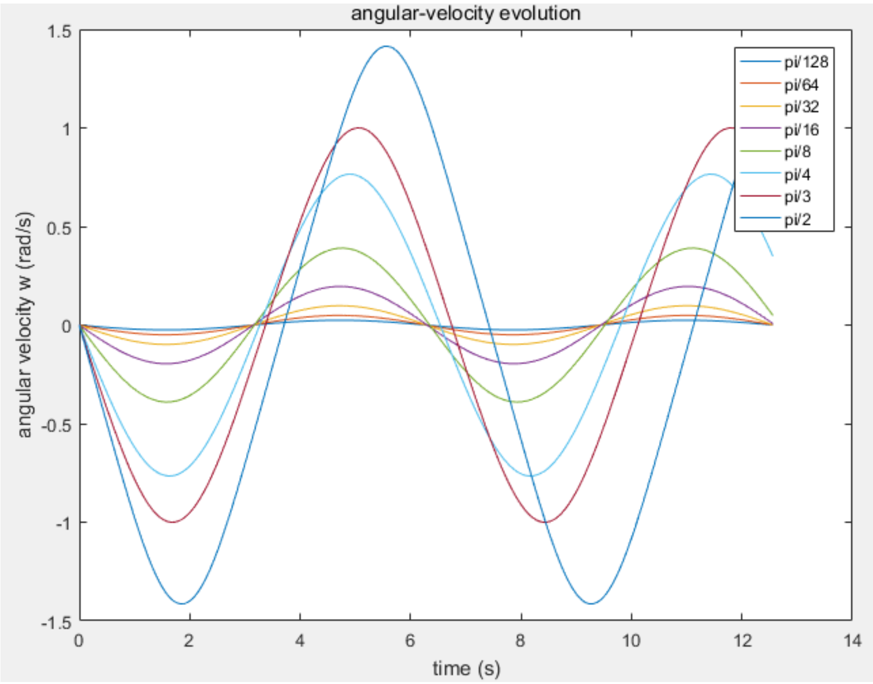
We firstly plot the angle-versus-time graph below. For comparison, we plot all graphs from different initial angles in one graph.



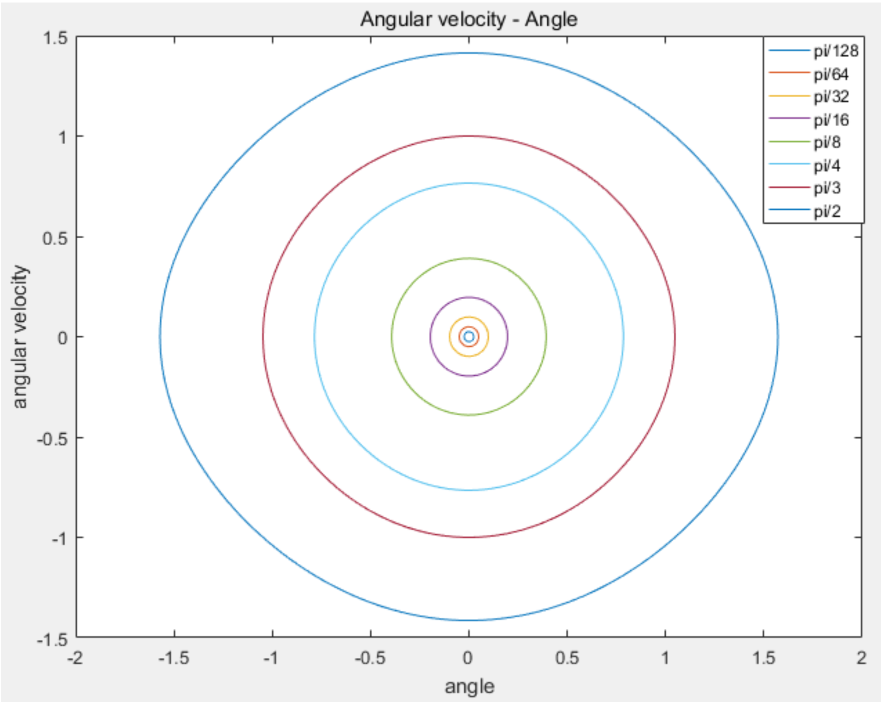


It comes out that, when the initial angles are small, i.e., pi/128, pi/64, pi/32, pi/16, the point arrives at the bottom roughly at the same time (see below). However, when initial angles are pi/8, pi/4, pi/3, pi/2, the time that the point arrives at the bottom becomes longer(it is not a matter of accuracy, since we get the same result with higher time resolution). It's clear when the initial angle is large, the motion no longer conforms to the small angle approximation.

And we plot the angular-velocity w versus time, the result shows the same essential characteristics.



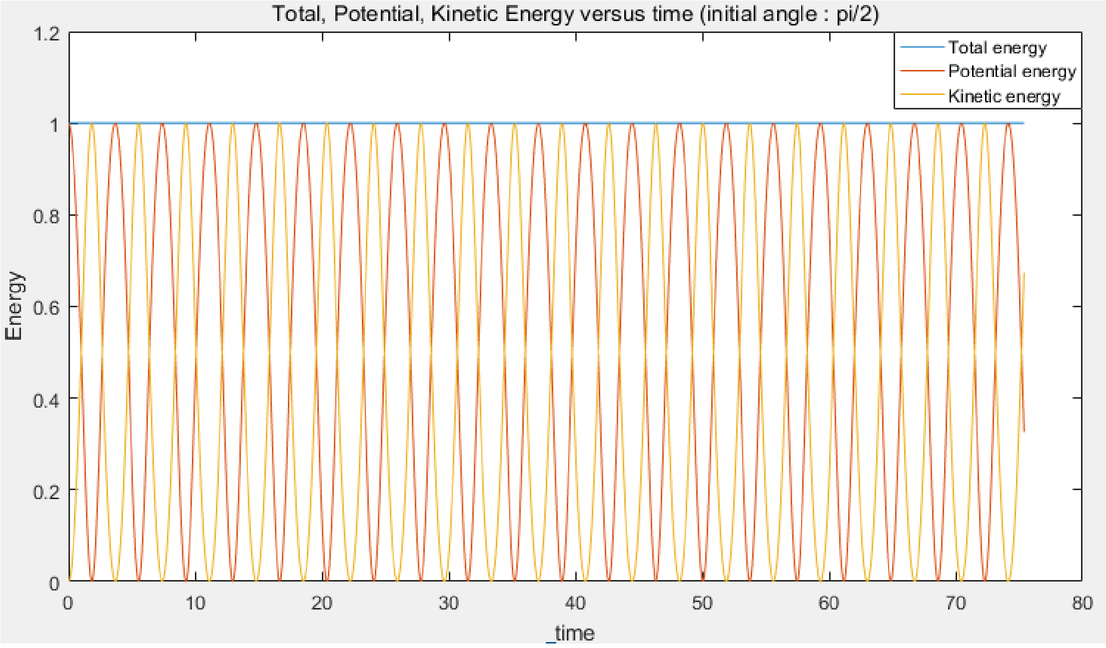
Then we plot the angular-velocity versus angle graph, we get



When initial angle is small, the curve conforms to a circle, and when initial angle is large, it shows a minute distortion.

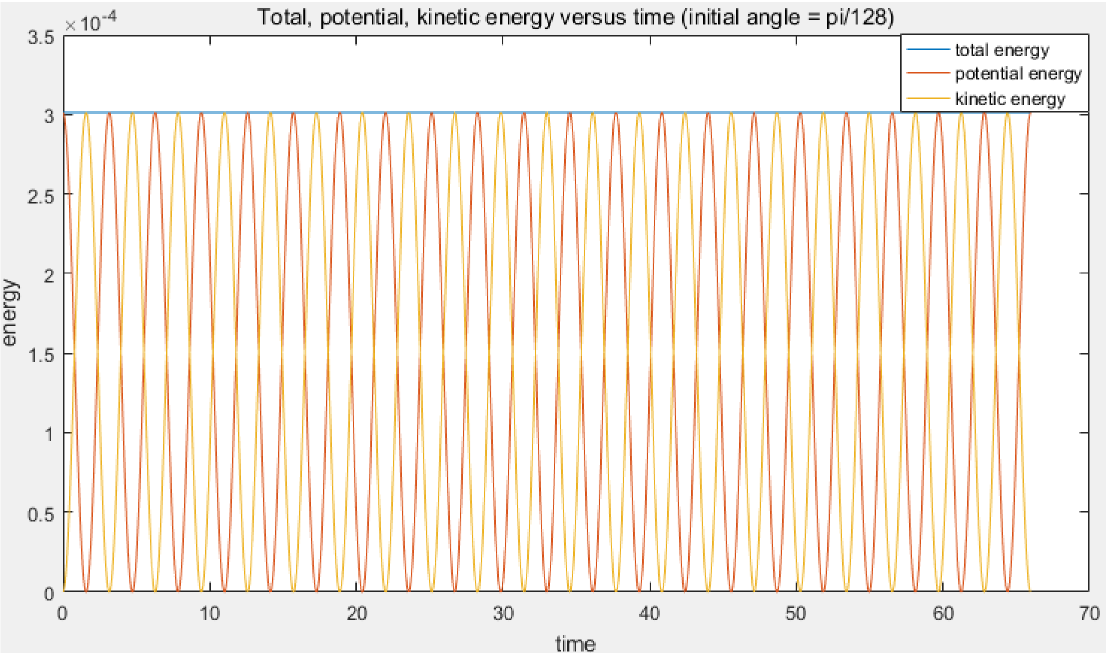
b) we set l=g=m=1. (step:0.01)

Firstly we choose initial angle = pi/2, we plot the total energy, potential energy, kinetic energy versus time in a single graph.

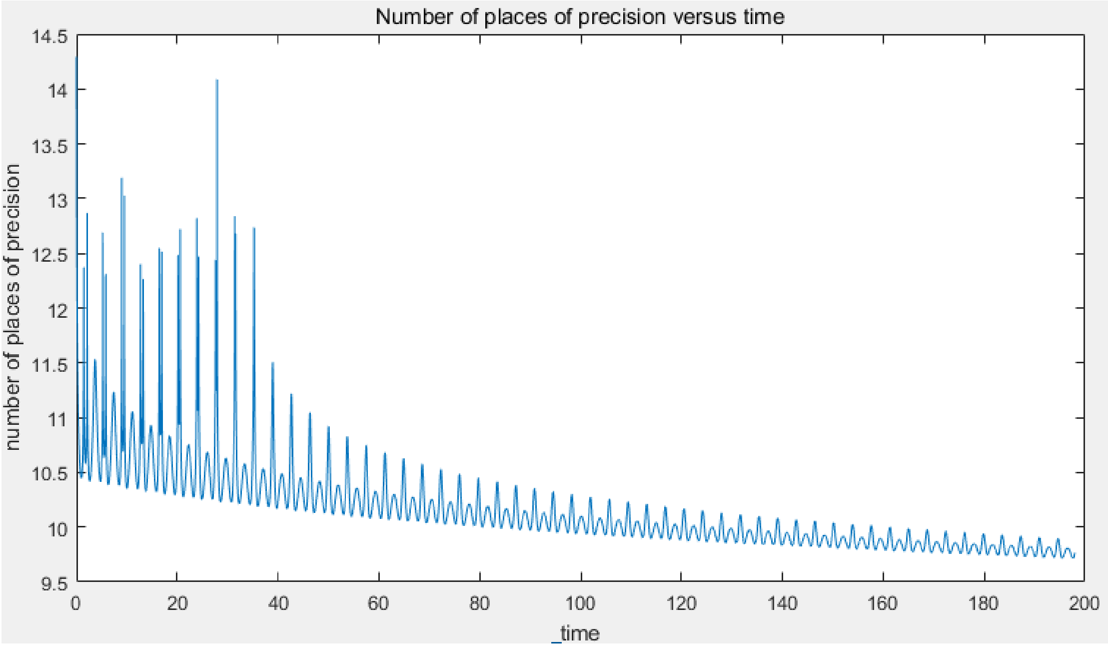


It shows that the total energy is conservative.

Then we choose the initial angle = pi/128, the result is similar.



c) as before we set l=g=m=1, we use step = 0.01, calculate about 30 periods. And here the initial angle is pi/2.

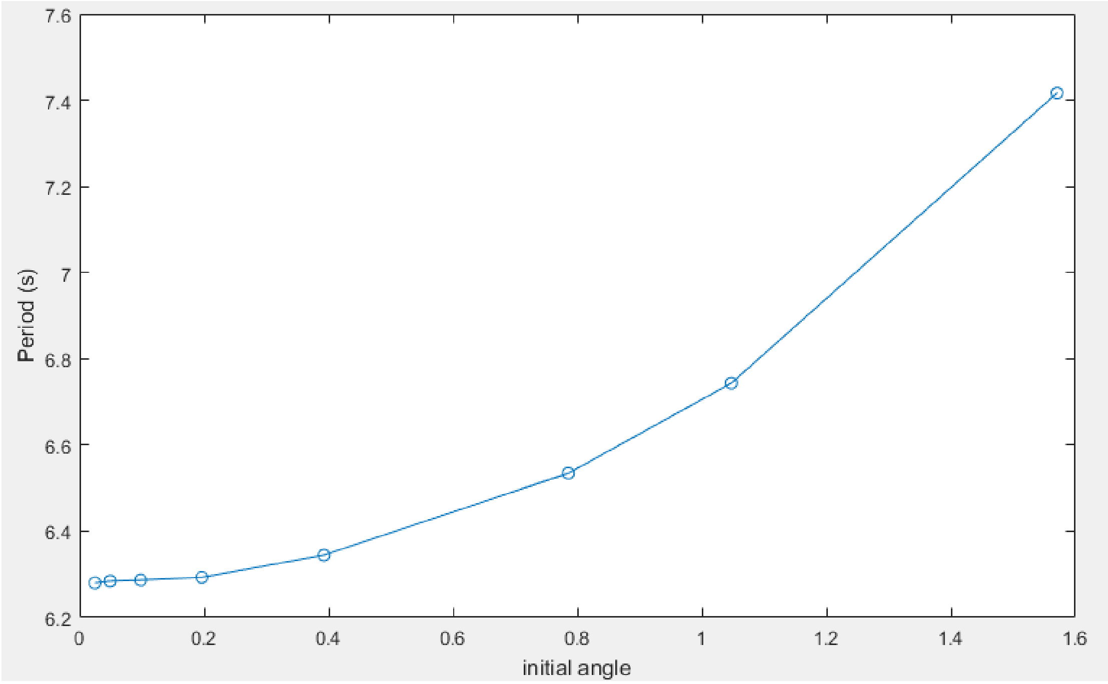


As time increases, the number of places of precision decreases gradually, also in an oscillation manner. In the time scale of our calculation, the number of places of precision is larger than 9 places.

d) we search for the moments at which the points arrives at its initial position (our approach: estimate the time range in which theta gets its local maximum, then find the maximum and its index, to get the arriving moments)

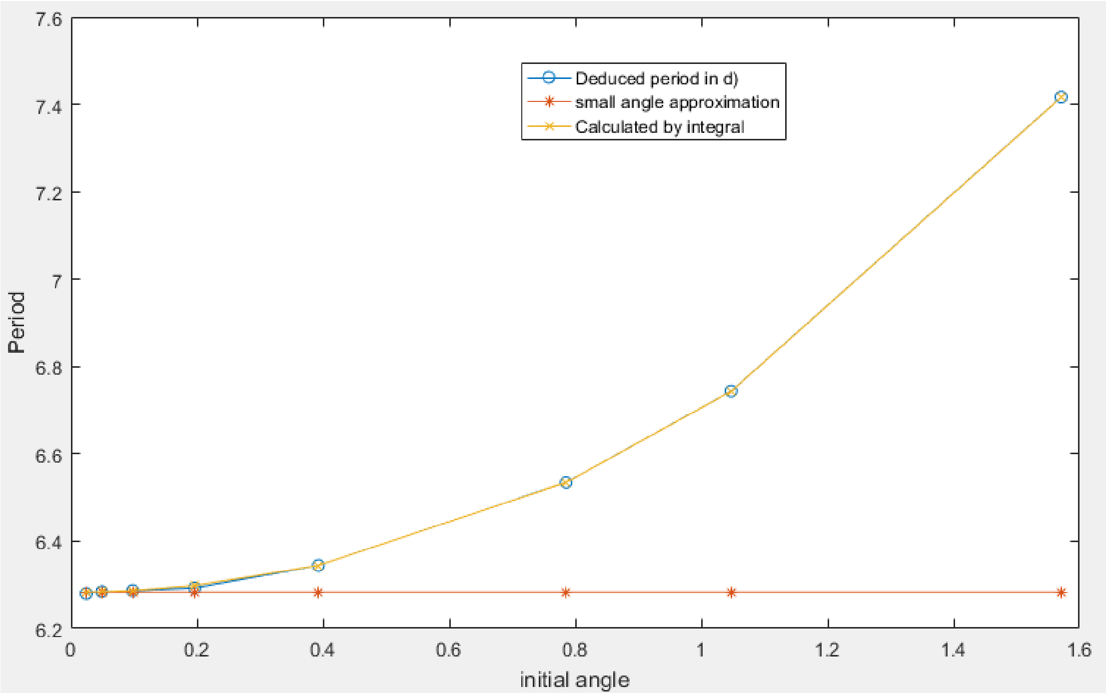
Initial angle	T1	T2	T3	T4	T5	T6	T (s) (=(T4+T5+T6-T1-T2-T3)/9)
Pi/128	0	6.28	12.57	18.85	25.13	31.42	6.28
Pi/64	0	6.28	12.57	18.85	25.14	31.42	6.2844
Pi/32	0	6.29	12.57	18.86	25.15	31.43	6.2867
Pi/16	0	6.30	12.60	18.90	25.19	31.49	6.2928
Pi/8	0	6.34	12.69	19.03	25.38	31.72	6.3444
Pi/4	0	6.53	13.07	19.60	26.14	32.67	6.5344
Pi/3	0	6.74	13.49	20.23	26.97	33.72	6.7433
Pi/2	0	7.42	14.83	22.25	29.67	37.08	7.4167

We plot the deduced period of a function of initial angle.



e) we list the results of small angle approximation, results in d), and results of integral as below.

Initial angle	Small angle approximation	Deduced in d)	integral
Pi/128	6.28	6.28	6.2834
Pi/64	6.28	6.2844	6.2841
Pi/32	6.28	6.2867	6.2870
Pi/16	6.28	6.2928	6.2984
Pi/8	6.28	6.3444	6.3443
Pi/4	6.28	6.5344	6.5343
Pi/3	6.28	6.7433	6.7430
Pi/2	6.28	7.4167	7.4163



The results in d) are consistent with the results from calculating integral.

Codes for Problem.2

RK4

```
function u = RK4ODESystem(f,x,u0)
u=zeros(length(u0),length(x));
u(:,1) = u0;
% The first column is set to be the initial vector u0
h = x(2) - x(1);
halfh=h/2;
for i = 1:length(x)-1,
    k1 = f(x(i),u(:,i));
    k2 = f(x(i)+halfh,u(:,i) + halfh*k1);
    k3 = f(x(i)+halfh,u(:,i) + halfh*k2);
    k4 = f(x(i)+h,u(:,i) + h*k3);
    u(:,i+1) = u(:,i)+h* (k1+2*k2+2*k3+k4)/6;
end
```

angle, angular velocity calculating scripts

```
f=inline('[y(2);-sin(y(1))]','t','y');
x=0:0.01:15*pi; %step 0.01, total time 15pi

u0128=[pi/128;0];
u064=[pi/64;0];
u032=[pi/32;0];
u016=[pi/16;0];
u08=[pi/8;0];
u04=[pi/4;0];
```

```
u03=[pi/3;0];
u02=[pi/2;0];
u128 = RK4ODESystem(f,x,u0128);
u64 = RK4ODESystem(f,x,u064);
u32 = RK4ODESystem(f,x,u032);
u16 = RK4ODESystem(f,x,u016);
u8 = RK4ODESystem(f,x,u08);
u4 = RK4ODESystem(f,x,u04);
u3 = RK4ODESystem(f,x,u03);
u2 = RK4ODESystem(f,x,u02);
```

Find the maximum script (example)

```
[M,I]=max(u2(1,502:902));
I+500
[M,I]=max(u2(1,1402:1600));
I+1400
[M,I]=max(u2(1,2202:2400));
I+2200
[M,I]=max(u2(1,2802:3000));
I+2800
[M,I]=max(u2(1,3602:3800));
I+3600
```