The commands needed to complete the above three assignments in Exercise 2 can be as follows.

```
s[x_,n_]:=NSum[(-1)^(k-1)x^(2k-1)/((2k-1)!),{k,1,n}];
figsin=Plot[Sin[x],{x,-2Pi,2Pi},
                        PlotStyle->{RGBColor[1,0,0]}];
figtylor2=Plot[s[x,2],{x,-3,3}];
                        PlotStyle->{RGBColor[1,0,1]}];
figtylor34=Plot[{s[x,3],s[x,4]},{x,-4,4];
figtylor5=Plot[{s[x,4],s[x,5]},{x,-5,5},
                        PlotStyle->{RGBColor[0,0,1]}];
Show[figsin,figtylor2,figtylor34,figtylor5]
```

* **Notes.** (i) The command `NSum[<the expression>,{k,1,n}]` has the same meaning as the mathematical symbol "$\sum_{k=1}^{n}$`<the expression>`", which means to calculate the sum of the terms obtained from `<the expression>` by taking $k = 1, 2, \ldots, n$.

(ii) The option "`PlotStyle->{RGBColor[1,0,0]}`" means to draw the graph red. The three independent variables in the function `RGBColor[r, g, b]` range from 0 to 1, representing the intensity of the three colors red, green and blue. So, `RGBColor[1,0,0]` means to draw the graph red, while `RGBColor[0,0,1]` means blue, `RGBColor[1,0,1]` means purple, etc. Since we draw many different curves in the same graph, it is good to use different colors to distinguish the curves.

```
f[x_,n_]:=Sum[Sin[k*x],{k,1,n,2}];
Plot[f[x,9],{x,-2Pi,2Pi}]
```

```
p[x_,n_]:=x*Product[1-x^2/((k*Pi)^2),{k,1,n}];
Plot[{Sin[x],p[x,5]},{x,-2Pi,2Pi}]
```

The `f[x,9]` in the commands above can be replaced by `f[x,19]` or `f[x,50]` or even `f[x,550]`, while `p[x,5]` can be replaced by `p[x,10],p[x,15]`.

1.2 The Number e

The logarithm we learned in high school is based on 10, which is called the **common logarithm** and denoted by $\lg N$. But the real commonly used logarithm in science is based on an irrational number $e = 2.71828\ldots$, called the **natural logarithm** and denoted by $\ln N$ or $\log N$. Why is the logarithm based on this eccentric irrational number e more natural than that based on 10? In 17th century when J. Napier invented logarithm, his objective was to simplify the calculation of astronomic data by changing multiplication into addition and changing division into subtraction. He wanted to describe each positive real number N as a power of a certain positive real number a: $N = a^n$. If $N = a^n$ and $M = a^m$, then $MN = a^{m+n}$. The multiplication of M and N becomes the addition of m and n. The problem is that first we must compile a logarithm form, list the corresponding relations between the power N (antilogarithm number) and the exponent n (logarithm). If it is based on 10, then the logarithm form might be:

N	1	10	100	1000	$\cdots$
$\lg N$	0	1	2	3	$\cdots$

This form does not seem to be good. For example, can you find the logarithm of 2 or 3 in it? The step between the adjacent antilogarithms in the form is too large! It jumps from 1 to 10 directly, without passing through $2, 3, \ldots, 9$. And then it jumps from 10 to 100, and then to 1000, the steps go larger and larger. In order to overcome this defect and make adjacent antilogarithms in the form closer, the base a should be near 1. Say $a = 1.001$. (In fact, Napier took $a = 0.99999$.)

Exercise 3. (1) Compile the logarithm form based on $a = 1.001$. List in the form the pairs of data $\{1.001^k, k\}$ for $k = 1, 2, \ldots, 2500$. The command in Mathematica is:

```
T=Table[{1.001^k,k},{k,1,2500}]
```

Use the form to find the approximation of $\lg 2$, by finding $b_1 = \log_{1.001}2$ and $b_2 = \log_{1.001}10$ and calculating their quotient $\lg 2 = b_1/b_2$.

In the form one cannot find a 1.001^k equal to 2 exactly. However, one can find

$$x_1 = 1.001^k < 2 < x_2 = 1.001^{k+1}\,.$$

One can take either k or $k+1$ as an approximation of $\log_{1.001}2$. However, if you hope to get a more accurate approximation, it is better to choose some y between k and $k+1$. Consider it in this way: from x_1 to x_2 we go with a step $x_2 - x_1$, while from x_1 to 2 we go with a step $2 - x_1$, which is a part of the ratio

$$u = \frac{2 - x_1}{x_2 - x_1}$$

of the whole step $x_2 - x_1$. We may assume the step from the corresponding logarithm k to $\log_{1.001} 2$ has about the same ratio u in the whole step $(k+1) - k$. Namely, we take

$$k + \frac{2 - x_1}{x_2 - x_1}$$

as an approximation of $\log_{1.001} 2$, which is certainly better than k and $k+1$. This method is called **linear interpolation**. Similarly one can use this method to find a good approximation of $\log_{1.001} 10$.

It is obvious that the logarithm form based on $a = 1.001$ obtained above is better than that based on 10. The only flaw is that the values of the logarithm calculated are too large. This flaw can be rectified easily by dividing all the values of logarithms in the form by the same constant. In view of $a = 1.001 = 1 + 1/1000$, it is natural to divide all the logarithms by 1000, and thus replace each $y = \log_{1.001} x$ by

$$y = \frac{\log_{1.001} x}{1000} = \frac{\log_{1.001} x}{\log_{1.001}(1.001^{1000})} = \log_b x\,,$$

where $b = 1.001^{1000}$. This means that we in fact obtain a logarithm form based on 1.001^{1000}. To increase the accuracy we may take 1.0001 instead of 1.001, and divide all the logarithms $\log_{1.0001} x$ obtained by 10000, to obtain a logarithm form based on 1.0001^{10000}. In general, we first take $1 + \frac{1}{n}$ to be the base of the logarithm, and then divide all the logarithms by n, to get a logarithm form based on $a_n = \left(1 + \frac{1}{n}\right)^n$, the larger the n, the better it will be. What happens when n tends to infinity?

Exercise 4. Observe the changing trend of the series $a_n = \left(1 + \frac{1}{n}\right)^n$ and $A_n = \left(1 + \frac{1}{n}\right)^{n+1}$ when n increases infinitely.

(1) Calculate the values a_n and A_n for $n = 10, 100, 1000, 10000$. Observe the changing trend. The Mathematica command is:

```
Do[Print[{1.0+10^(-m))^(10^m),(1.0+10^(-m))^(10^m+1)}],
                                              {m,1,4}]
```

(2) Construct the following three functions in the same coordinate system:

$$y = (1 + 10^{-x})^{10^x}, \quad y = (1 + 10^{-x})^{10^x + 1}, \quad y = e\,.$$

Observe the trend of these graphs when x increases. The construction command in Mathematica is:

```
Plot[{(1+10^(-x))^(10^x),(1+10^(-x))^(10^x+1),E},{x,1,4}]
```

The graph that the command realizes describes the function in $[1, 4]$. You may try other intervals such as $[2, 5]$ or $[5, 7]$, and observe the results.

One sees that when n increases, a_n also increases but A_n decreases, and a_n and A_n get closer and closer. When n goes large enough, a_n and A_n tend to the same limit $e = 2.71828\ldots$. This is just the base of the **natural logarithm**. (Of course, all these conclusions can be proved by theoretical deduction. One can refer to calculus textbooks for the proof.)

(3) Use the following command to calculate the approximation of

$$e = 1 + \sum_{k=1}^{\infty} \frac{1}{k!}$$

to the 20th (or more) decimal places.

```
Do[Print[N[1.0+NSum[1/(k!),{k,1,n}],20],{n,10,20}]]
```

(*where `N[<the expression>,20]` assign the real output to the accuracy up to the 20th decimal place).

The content above is to describe how the natural logarithm turns out naturally in compiling the logarithm form. When Napier compiled his logarithm form he might not put forward the terminology "natural logarithm", but his earliest logarithm form was certainly not the "common logarithm" form at all. His logarithm based on 0.99999 is essentially the "natural logarithm" up to a constant multiple. However, for the convenience of the use by people he changed the base of the logarithm, compiled the "common logarithm" form based on 10.

The more direct reason that the natural logarithm based on e is commonly used in science is: It makes the differential or integral formulae with logarithm involved become the simplest.

Exercise 5. (1) Run the following Mathematica command to calculate the values of $u(x) = \frac{\log_{10}(1+x)}{x}$ for $x = 0.1, 0.01, 0.001, \ldots, 10^{-7}$ etc.

```
Do[Print[Log[10,1+10^(-m)]/(10^(-m))],{m,1,7}]
```

(*In Mathematica, `Log[a,N]` represents the logarithm of N based on a, so `Log[10,N]` represents the common logarithm $\lg N$. When we write `Log[N]` with the base omit, it means that the base is e and `Log[N]` represents the natural logarithm. The loop command `Do[<circulation body>,{n,1,7}]` represents operating the loop body orderly, for n from 1 to 7. In this example the loop body is `Print[<the expression>]`, whose function is to print the value of `<the expression>`.)

Observe whether $u(x)$ tends to a certain limit u when x is close to 0. This limit u is just the derivative of the common logarithm $y = \lg x$ at $x = 1$. One may see that it is not a simple number. One may agree that if this limit is equal to 1, it would be more pleasant. This can be easily done: divide the function $\lg x$ by u, replace it by $(\lg x)/u$, then its derivative at 1 will be 1. However,

$$\frac{\lg x}{u} = \frac{\lg x}{\lg(10^u)} = \log_{10^u} x$$

is just the logarithm based on 10^u.

(2) Calculate the value of 10^u. Is it a value you are familiar with?

(3) Calculate the values $v(x) = \frac{\ln(1+x)}{x}$ for $x = 10^{-n}$, with $n = 1, 2, \ldots, 7$. The Mathematica command is

```
Do[Print[Log[1+10^(-m)]/(10^(-m))],{m,1,7}]
```

Observe whether $v(x)$ tends to a certain limit when x trends to 0. What is the value of this limit?

1.3 Integral and Natural Logarithm

For a positive real number a, study the area of $S(a)$ which is bounded by the graph of the inverse function $y = \frac{1}{x}$, the x-axis and the straight lines $x = 1$ and $x = a$.

This area is just the value of the definite integral $\int_1^a \frac{1}{x} dx$. Notice that we define the straight line $x = 1$ to be the reference of calculating the area. The area to the right of this line is positive, and to the left is negative. That is to say, when $a > 1$, $S(a) > 0$; when $0 < a < 1$, $S(a) < 0$; when