Question 1:

```
#include <iostream>
using namespace std;
class hiding{
private:
    int num;
    char ch;
public:
    void set(int n, char c) {
        num = n;
        ch = c;
    }
    void get() {
        cout<<"Numbers is: "<<num<< endl;
        cout<<"Char is: "<<ch<<endl;
    }
};
int main(){
    hiding obj;
    obj.set(100, 'X');
    obj.get();
    return 0;
}
```

Practice 1:

```
#include <iostream>
using namespace std;
class hiding{
private:
    int roll;
    char ch;
    int marks;
public:
    void set(int r, char c) {
        roll = r;
        ch = c;
    }
    void get() {
        cout<<"Roll No.: "<<roll<< endl;
        cout<<"Name: "<<ch<<endl;
    }
    void grade(int per){
        marks = per;
        if(per<=100&&per>=80)
            cout<<"Grade: A";
        else if(per<80&&per>=70)
            cout<<"Grade: B";
        else if(per<70&&per>=50)
            cout<<"Grade: C";
        else
            cout<<"Grade: F";
    }
};
int main(){
    hiding obj;
    obj.set(100, 'X');
    obj.get();
    obj.grade(92);
    return 0;
}
```

Question 2:

```
#include <iostream>
using namespace std;
class Sum{
    private:
        int x, y, z;
    public:
    void add(){
        cout<<"Enter two numbers: ";
        cin>>x>>y;
        z= x+y;
        cout<<"Sum of two number is: "<<z<<endl;
    }
};
int main(){
    Sum sm;
    sm.add();
    return 0;
}
```

Practice 2:

```
#include <iostream>
```

```
using namespace std;
class Largest{
    private:
        int x, y, z;
    public:
        int num1, num2, num3;
    void set(){
        cout<<"Enter three numbers: "<<endl;
        cin>>x>>y>>z;
        num1 = x;
        num2 = y;
        num3 = z;
    }
    void get(){
        cout<<"First number: "<<num1<<endl;
        cout<<"Second number: "<<num2<<endl;
        cout<<"Three number: "<<num3<<endl;
    }
    void maxnum(){
        if(num1 > num2 && num1 > num3){
            cout<<"First number is the largest";
        }
        else if(num2 > num1 && num2 > num3){
            cout<<"Second number is the largest";
        }
        else{
            cout<<"Third number is the largest";
        }
    }
};
int main(){
    Largest ls;
    ls.set();
    ls.get();
    ls.maxnum();
    return 0;
}
```

Question 3:

```
#include <iostream>
#include<math.h>
using namespace std;
int main()
```

```
{
   int n = 4;
   int power = 3;
   int result = pow(n,power);
   cout << "Cube of n is : " <<result;
   return 0;
}
```

Practice 3:

```
#include <iostream>
#include <math.h>
using namespace std;

void checkperfectsquare(int n)
{
    if (ceil(sqrt(n)) == floor(sqrt(n))) {
        cout << "Perfect square";
    }
    else {
        cout << "Not a perfect square";
    }
}

int main()
{
    int n = 50;
    checkperfectsquare(n);
    return 0;
}
```

Question 4:

An interface describes the behavior or capabilities of a C++ class without committing to a particular implementation of that class.

The C++ interfaces are implemented using abstract classes and these abstract classes should not be confused with data abstraction which is a concept of keeping implementation details separate from associated data.

```
#include <iostream>
using namespace std;
```

```
class Shape {
  public:
   // pure virtual function providing interface framework.
   //A pure virtual function is a virtual function in C++
   //for which we need not to write any function definition and only we have to declare it.
   //It is declared by assigning 0 in the declaration.
   virtual int getArea() = 0;
   void setWidth(int w) {
     width = w;
   }

   void setHeight(int h) {
     height = h;
   }

   protected:
     int width;
     int height;
};

class Rectangle: public Shape {
  public:
    int getArea() {
      return (width * height);
    }
};

class Triangle: public Shape {
  public:
    int getArea() {
      return (width * height)/2;
    }
};

int main(void) {
  Rectangle Rect;
  Triangle  Tri;

  Rect.setWidth(5);
  Rect.setHeight(7);

  cout << "Total Rectangle area: " << Rect.getArea() << endl;

  Tri.setWidth(5);
```

```
    Tri.setHeight(7);

    cout << "Total Triangle area: " << Tri.getArea() << endl;

    return 0;
}
```