An agency needs to keep information regarding Real Estate for sale. They sell Lands, houses and buildings
Design and implement the class(es) to fulfill the following data.

- Dimensions…The length and the width in meters. (integer)
- City Tax… amount in $ (float)
- Selling price… amount in $ (float)
- A house has:
  - Number of rooms. (integer)
  - School tax… amount in $ (float)
- A building has:
  - Number of floor. (integer)
  - Water tax… amount in $ (float)
- A Land has:
  - Zone category… a character string indicating the zoning category of the land (Green, Industry, Tourist…)

Create the class(es) and all the needed constructors, getters and setters. Use your judgment for the defaults arguments.
You should also implement a ***printInfo*** function that prints all the information regarding a real estate object. This information includes the total taxes.


```
#include <iostream.h>
#include <iomanip.h>
#include <string.h>

class RealEstate {
 public:
   RealEstate(int ln=0, int wd=0, float cT=0.0, float prc=0.0) {
      length = ln >0 ? ln : 0;
       width = wd >0 ? wd : 0;
       cityTax = cT >0.0 ? cT : 0.0;
       price = prc >0.0 ? prc : 0.0;
   };

   virtual void printInfo() {
      cout << "Dimension: "<< endl
           << "\tLength= " << length << " meters \n"
           << "\twidth= " << width << " meters \n"
           << "Price: $ " << price <<endl;
   }

 protected:
   int length;
   int width ;
   float cityTax;
   float price;
};
```



```
class House : public RealEstate {
 public:
   House (int ln, int wd, float cT, float prc, int rm, float sT)
                 : RealEstate(ln, wd, cT, prc) {
      nbRoom = rm >0 ? rm : 0;
      schoolTax = sT >0.0 ? sT : 0.0;
   }

   void printInfo() {
      cout << "Its is a " << nbRoom << " Rooms House\n";
      RealEstate::printInfo();
      cout << "Taxes are evaluated at $ "
           << schoolTax + cityTax
           << " per year \n";
   }

 protected:
   int nbRoom;
   float schoolTax;
};
```



```
class Building : public RealEstate {
 public:
   Building (int ln, int wd, float cT, float prc, int fl, float wT)
                 : RealEstate(ln, wd, cT, prc) {
      nbFloor = fl > 0 ? fl : 0;
      waterTax = wT >0.0 ? wT : 0.0;
   }

   void printInfo() {
      cout <<"Its is a " <<nbFloor <<" Floor Building\n";
      RealEstate::printInfo();
      cout << "Taxes are evaluated at $ "
           << waterTax + cityTax
           << " per year \n";
   }

 protected:
   int nbFloor;
   float waterTax;
};
```

```
class Land : public RealEstate {
 public:
   Land (int ln, int wd, float cT, float prc, char *zn)
                  : RealEstate(ln, wd, cT, prc) {
      strncpy(zone,zn,9);
      zone[9]='\0';
   }

   void printInfo() {
      cout << "Its is a " << zone << " Zone Land\n";
      RealEstate::printInfo();
      cout << "Taxes are evaluated at $ "
           << cityTax << " per year \n";
   }

 protected:
   char zone[10];
};
```



```
void main () {
   RealEstate *ReMax[20];

   cout << setiosflags( ios::fixed | ios::showpoint )
        <<  setprecision( 2 );

   ReMax[0]= new House(10,15,1000.0 , 80000.0 , 5, 200.0);
   ReMax[0]->printInfo();

   cout << endl;

   ReMax[1]=new Building(20,30,4300.0, 250000.0, 3, 1200.0);
   ReMax[1]->printInfo();

   cout << endl;

   ReMax[2] = new Land(15,18,500.0 , 7000.0 , "Green");
   ReMax[2]->printInfo();


}
```