

```

1  /* 2.Cree una clase base llamada Edificio, cree 3 clases
•  derivadas de la clase Edificio,
2      llamadas Residencial, Comercial, Educativo. Haga uso del
•  polimorfismo incluyendo la
3      función virtual mostrarEdificio(); en la clase Edificio esta
•  función solo mostrara la
4      Ubicación y Color, pero en las demás clases deberá imprimir
•  al menos un dato específico
5      de la clase derivada Ejemplo: en Residencial,
•  numeroDeResidentes. */
6  #include <iostream>
7  #include <string>
8
9  using namespace std;
10
11  class Edificio{
12  public:
13
14      void set_color(){
15          cout << "Introduce el color: ";
16          cin >> color;
17      }
18      void set_ubicacion(){
19          cout << "Introduce la ubicacion: ";
20          cin >> ubicacion;
21      }
22      string get_color(){
23          return color;
24      }
25      string get_ubicacion(){
26          string x = ubicacion;
27          return x;
28      }
29      virtual void mostrarEdificio(){
30          cout << "El color del Edificio es " << get_color();
31          cout << "La ubicacion del Edificio es " << get_ubicacion();
32      }
33  private:
34      string color;
35      string ubicacion;
36
37  };
38
39  class Residencial: public Edificio{
40  public:
41
42      void set_numeroResidentes(int x){
43          numeroResidentes = x;
44      }
45      void set_numHabitaciones(int x){
46          numHabitaciones = x;

```

```

47     }
48     void set_seguridad(bool x){
49         seguridad = x;
50     }
51     void set_gimnasio(bool x){
52         gimnasio = x;
53     }
54     void set_plusvalia(bool x){
55         plusvalia = x;
56     }
57
58     void mostrarEdificio(){
59
60         cout << "La residencia color " << get_color();
61         cout << " ubicada en " << get_ubicacion();
62         cout << " cuenta con " << numHabitaciones << " habitaciones"
63             << ", el numero maximo de residentes es " <<
        *         numeroResidentes << endl;
64         seguridad == true ? cout << "La residencia cuenta con
        *         seguridad privada" : cout << "La residencia no cuenta con
        *         seguridad privada";
65         gimnasio == true ? cout << ", gimnasio propio" : cout <<
        *         ", no cuenta con gimnasio propio";
66         plusvalia == true ? cout << ", esta bien ubicada.\n" : cout
        *         << ", se encuentra en zona de desarrollo.\n";
67     }
68 private:
69     int numeroResidentes;
70     int numHabitaciones;
71     bool seguridad;
72     bool gimnasio;
73     bool plusvalia;
74
75 };
76 class Educativo: public Edificio{
77 public:
78     void set_alumnosPorSalon(int x){
79         alumnosPorSalon = x;
80     }
81     int get_alumnosPorSalon(){
82         return alumnosPorSalon;
83     }
84     void set_cicloEscolar(string x){
85         cicloEscolar = x;
86     }
87     string get_cicloEscolar(){
88         return cicloEscolar;
89     }
90     void set_tipoEscuela(string x){
91         tipoEscuela = x;
92     }

```

```

92     }
93     string get_tipoEscuela(){
94         return tipoEscuela;
95     }
96     void mostrarEdificio(){
97         cout << "La " << get_tipoEscuela() << " color " <<
98         * get_color();
99         cout << " ubicada en " << get_ubicacion();
100        cout << " Por salon hay una cantidad de " <<
101        * get_alumnosPorSalon() << " alumnos" << endl;
102        cout << "El ciclo escolar es " << get_cicloEscolar() <<
103        * endl;
104    }
105 private:
106     int alumnosPorSalon;
107     string cicloEscolar;
108     string tipoEscuela;
109 };
110
111 class Comercial: public Edificio{
112 public:
113     void set_renta(double x){
114         renta = x;
115     }
116     void set_proteccion(bool x){
117         x = proteccion;
118     }
119     double get_renta(){
120         return renta;
121     }
122     bool get_proteccion(){
123         return proteccion;
124     }
125     void mostrarEdificio(){
126         cout << "La residencia color " << get_color();
127         cout << " ubicada en " << get_ubicacion();
128         cout << " renta mensual $" << renta << " mxn.";
129         * proteccion == true ? cout << " El comercio cuenta con
130         * proteccion" : cout << " El comercio no cuenta con
131         proteccion";
132     }
133 private:
134     double renta;
135     bool proteccion;
136 };
137
138 int main(){

```

```
138     Residencial casa1;
139     casa1.set_color();
140     casa1.set_ubicacion();
141     casa1.set_numHabitaciones(4);
142     casa1.set_numeroResidentes(32);
143     casa1.set_gimnasio(false);
144     casa1.set_plusvalia(true);
145     casa1.set_seguridad(true);
146     casa1.mostrarEdificio();
147
148     Comercial comercio1;
149     comercio1.set_color();
150     comercio1.set_ubicacion();
151     comercio1.set_proteccion(false);
152     comercio1.set_renta(2500);
153     comercio1.mostrarEdificio();
154
155     Educativo primaria;
156     primaria.set_color();
157     primaria.set_ubicacion();
158     primaria.set_alumnosPorSalon(26);
159     primaria.set_cicloEscolar("2018-2019");
160     primaria.set_tipoEscuela("primaria");
161     primaria.mostrarEdificio();
162
163     return 0;
164 }
165
```

Introduce el color: azul

Introduce la ubicacion: centro

La residencia color azul ubicada en centro cuenta con 4 habitaciones, el numero maximo de residentes es 32

La residencia cuenta con seguridad privada, no cuenta con gimnasio propio, esta bien ubicada.

Introduce el color: verde

Introduce la ubicacion: norte

La residencia color verde ubicada en norte renta mensual \$2500 mxn. El comercio no cuenta con proteccion

Introduce el color: naranja

Introduce la ubicacion: sur

La primaria color naranja ubicada en sur Por salon hay una cantidad de 26 alumnos

El ciclo escolar es 2018-2019