Source code:

Person.h file:

```
#ifndef _PERSON_H_
#define _PERSON_H_
#include <string>
using namespace std;

class Person{
   private:
        string name;
        double salary;

   public:
        void setPerson();
        string getName();
        double getSalary();
};
#endif //_PERSON_H
```

Person.cpp file:

```
#include "Person.h"

void Person::setPerson(){
   cout << "Enter name: ";
   cin >> this -> name;
   cout << "Enter Salary: ";
   cin >> this -> salary;
   cout << endl;
}

string Person::getName(){
   return this->name;
}

double Person::getSalary(){
   return this->salary;
}
```

Main.cpp file:

```
#include <iostream>
#include "Person.cpp"
using namespace std;

void bSort(Person **person, int n, bool s);
void order(Person **personi, Person **personiPlus1);
void display(Person **person, int length);

int main(){
   int numOfPeople {0};
   cout << "How many people would you like to enter? ";
   cin >> numOfPeople;
   cout << endl;

   Person *ptr[numOfPeople];// have array of pointers of type person

   for (int i {0}; i < numOfPeople; i++){//fill up array of type person
       ptr[i] = new Person();
       ptr[i] -> setPerson();
   }

   cout << "Unsorted Order of People: " << endl;
   display(ptr, numOfPeople);
   cout << endl;

   cout << "People Sorted by Salary: " << endl << endl;
   bSort(ptr, numOfPeople, false);
   display(ptr, numOfPeople);
   cout << endl;

   cout << "People Sorted by Name: " << endl << endl;
   bSort(ptr, numOfPeople, true);
   display(ptr, numOfPeople);
   cout << endl;

   for (int i {0}; i<numOfPeople; i++){ //deleting the array of pointers
       delete ptr[i];
   }
   return 0;
```

```
}

void bSort(Person **person, int n, bool s){
  for(int i {0}; i < n; i++){
      for(int j {i+1}; j < n; j++){
          Person *temp;
          if((s == false && person[i]->getSalary() > person[j] ->
getSalary()) || (s == true && person[i] -> getName() > person[j] ->
getName())){
           //    temp = person[i];
           //    person[i] = person[j];
           //    person[j] = temp;
           order(person + i, person + j);
          }
      }
  }
}

void order(Person **personi, Person **personiPlus1){
   Person *tmp = *personi;
   *personi = *personiPlus1;
   *personiPlus1 = tmp;
}

void display(Person **person, int length){
   for(int i {0}; i < length; i++){
       printf("%-10s$%-10.2f", person[i]->getName().c_str(),
person[i]->getSalary());
       cout << endl;
   }
}
```

Runtime output:
How many people would you like to enter? 3

Enter name: raymond
Enter Salary: 32234

Enter name: raul
Enter Salary: 54393

Enter name: alex
Enter Salary: 54589

Unsorted Order of People:
raymond $32234.00
raul $54393.00
alex $54589.00

People Sorted by Salary:

raymond $32234.00
raul $54393.00
alex $54589.00

People Sorted by Name:

alex $54589.00
raul $54393.00
raymond $32234.00