Q1. Write a c++ code, to demonstrate the forward list.

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

int main()
{
   forward_list <int> f;
   int e = 0;

   cout<<"Enter 5 elements in forward list = "<<endl;

   for(int i = 0; i < 5; i++)
   {
      cout<<i<<") = ";
      cin>>e;
      f.push_front(e);
   }

   cout<<"Display forward list"<<endl;
   forward_list <int> ::iterator it;
   it = f.begin();

   for(int i = 0; i < 5; i++, it++)
   {
      cout<<i<<") "<<*it<<endl;
   }

   cout<<endl<<"Pop element from forward list"<<endl;
   for(it = f.begin(); it != f.end(); it = f.begin())
   {
      cout<<*it<<" ";
      f.pop_front();
   }
}
```

Q2. Write a c++ code, in which create a forward list and assign values to it at the time of initialization and print it on the console screen.

```
#include <iostream>
#include <forward_list>

using namespace std;

int main()
{
   forward_list <int> f ={10, 20, 30, 40, 50};

   for(auto it = f.begin(); it != f.end(); it++)
   {
      cout<<*it<<" ";
   }

}
```

Q3. Create a forward list insert elements from 1 to 10 and find the sum of elements.

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

int main()
{
   forward_list <int> f = {10, 9, 8, 7, 6, 5, 4, 3, 2, 1};

   f.sort();

   int sum = 0;

   forward_list <int>::iterator it = f.begin();

   while( it != f.end())
   {
      sum = sum + (*it);
      it++;
   }

   cout<<"sum = "<<sum;
}
```

Q4. Write a program to reverse forward list elements.

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

int main()
{
   forward_list <int> f ={10, 20, 30, 40, 50};

   f.reverse();

   for(auto it = f.begin(); it != f.end(); it++)
   {
      cout<<*it<<" ";
   }
}
```

Q5. Write a program remove all consecutive duplicate elements from the forward list.

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

int main()
{
   forward_list <int> f;

   f.assign({10,20,30,20,40,40,20,50});

   cout<<"Before removing duplicate elements"<<endl<<endl;

   for(auto it: f)
   {
      cout<<it<<" ";
   }

   cout<<endl<<endl<<"After removing duplicate elements"<<endl<<endl;

   f.remove(20);
   f.remove(40);

   for(auto it: f)
   {
      cout<<it<<" ";
   }

}
```

Q6. Create two forward lists of int type, and merge them.

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

int main()
{
   forward_list <int> f1 = {10, 20, 30, 40, 50};
   forward_list <int> f2 = {60, 70, 80, 90, 100};

   f1.merge(f2);

   for(int x: f1)
   {
      cout<<x<<" ";
   }
}
```

Q7. Below are two forward lists, first sort them and then merge them.
forwardlist1={3,2,9}
forwardlist2={8,1,2}

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

int main()
{
   forward_list <int> forwardlist1 = {3, 2, 9};
   forward_list <int> forwardlist2 = {8,1,2};

   forwardlist1.sort();
   forwardlist2.sort();

   forwardlist2.merge(forwardlist1);

   for(int x: forwardlist2)
   {
      cout<<x<<" ";
   }
}
```

Q8. Create two forward lists of int type, and swap the elements of both forward lists with each other.

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

int main()
{
   forward_list <int> f1 = {1, 2, 3};
   forward_list <int> f2 = {10, 20, 30};

   cout<<"Before swapiing"<<endl<<endl;

   cout<<"f1 = ";
   for(forward_list <int>::iterator x = f1.begin(); x != f1.end(); x++)
   {
      cout<<*x<<" ";
   }

   cout<<endl<<"f2 = ";
   for(auto x : f2)
   {
      cout<<x<<" ";
   }

   cout<<endl<<endl<<"After swapping"<<endl;

   f1.swap(f2);

   cout<<endl<<"f1 = ";
   for(auto x : f1)
   {
      cout<<x<<" ";
   }

   cout<<endl<<"f2 = ";
   for(auto x : f2)
   {
      cout<<x<<" ";
   }
}
```

Q9. Write a C++ code to demonstrate working of splice_after() in forward list.

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

int main()
{
   forward_list <int> f1 = {10, 20, 30, 40, 50};
   forward_list <int> f2 = {1, 2, 3};



   f1.splice_after(f1.before_begin(), f2);


   for(int x : f1)cout<<x<<" ";
}
```

Q10. Write a program to assign values in forward_list using the values of another list

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

int main()
{
   forward_list <int> f1 = {1, 2, 3};
   forward_list <int> f2;

   f2.assign(f1.begin(), f2.end());

   for(int x : f2) cout<<x<<" ";
}
```