**PROGRAM**:

#include<iostream>

using namespace std;

int size;

struct Queue

{

int que[100];

int rear,front;

}q;

class queue

{

public:

int Qfull(int);

int insert(int);

int insert\_new(int);

int Qempty();

int Delete();

int display();

};

int queue::Qfull(int size)

{

if(q.rear>=size)

{

cout<<"\nQUEUE IS FULL!!NO MORE JOBS!!\n";

return 1;

}

else

{

return 0;

}

}

int queue::Qempty()

{

if(q.front==-1 || q.front>q.rear)

return 1;

else

return 0;

}

int queue::insert(int element)

{

while(Qfull(size)!=1)

{

if(q.front==-1)

q.front++;

q.que[++q.rear]=element;

//cout<<"\n"<<q.rear;

cout<<"\nTHANK YOU...your id is registered!\n";

return q.rear;

}

}

int size1;

int queue::insert\_new(int element)

{

while(Qfull(size+size1)!=1)

{

size+=size1;

if(q.front==size-1)

q.front++;

q.que[++q.rear]=element;

cout<<"\nTHANK YOU...your id is registered!\n";

return q.rear;

}

}

int item;

int queue::Delete()

{

if(Qempty()!=1)

//cout<<"enter the job no u want to delete\n";

//cin>>item;

item=q.que[q.front];

q.front++;

cout<<"\nALL ID'S ARE DELETED!!\n";

return q.front;

}

int queue::display()

{

for(int i=q.front+1;i<=q.rear;i++)

{

cout<<q.que[i]<<"\n";

}

}

int main()

{

char choice;

queue obj;

int ans;

int elem,ele;

cout<<"\nenter the total number of jobs\n";

cin>>size;

do

{

//cout<<"\ninsertion start\n";

cout<<"\npress 1 to enter elements:\npress 2 to display ID'S\npress 3 to delete the ID'S\npress 4 to enter new job id's\n";

cin>>ans;

switch(ans)

{

case 1:{

for(int i=0;i<size;i++)

{

cout<<"\nenter your job id:";

cin>>elem;

obj.insert(elem);

}

break;

}

//cout<<"\ninsertion end\n";

case 2:{

/\* for(int i=0;i<size;i++)

{

cout<<"\nenter your job id:";

cin>>elem;

obj.insert(elem);

obj.insert(elem);\*/

cout<<"\n------JOB ID'S OF EMPLOYEES ARE-------\n";

obj.display();

break;

}

case 3:{

obj.Delete();

break;

}

case 4:{

cout<<"\nenter number of new job id's you want to insert\n";

cin>>size1;

for(int j=0;j<size1;j++)

{

cout<<"\nenter the new job id you want to insert \n";

cin>>ele;

obj.insert\_new(ele);

}

cout<<"\n---------NEW LIST OF JOB ID'S ----------'\n";

obj.display();

break;

}

default:cout<<"\nINVALID CHOICE..try again\n";

}

cout<<"\ndo you want to continue using the system?(y/n):";

cin>>choice;

}while(choice=='y' || choice =='Y');

}

**OUTPUT:**

enter the total number of jobs

3

press 1 to enter elements:

press 2 to display ID'S

press 3 to delete the ID'S

press 4 to enter new job id's

1

enter your job id:1

THANK YOU...your id is registered!

enter your job id:2

THANK YOU...your id is registered!

enter your job id:3

THANK YOU...your id is registered!

do you want to continue using the system?(y/n):y

press 1 to enter elements:

press 2 to display ID'S

press 3 to delete the ID'S

press 4 to enter new job id's

2

------JOB ID'S OF EMPLOYEES ARE-------

1

2

3

do you want to continue using the system?(y/n):y

press 1 to enter elements:

press 2 to display ID'S

press 3 to delete the ID'S

press 4 to enter new job id's

4

enter number of new job id's you want to insert

2

enter the new job id you want to insert

4

THANK YOU...your id is registered!

enter the new job id you want to insert

5

THANK YOU...your id is registered!

---------NEW LIST OF JOB ID'S ----------'

1

2

3

4

5

do you want to continue using the system?(y/n):y

press 1 to enter elements:

press 2 to display ID'S

press 3 to delete the ID'S

press 4 to enter new job id's

4

enter number of new job id's you want to insert

1

enter the new job id you want to insert

10

THANK YOU...your id is registered!

---------NEW LIST OF JOB ID'S ----------'

1

2

3

4

5

10

do you want to continue using the system?(y/n):y

press 1 to enter elements:

press 2 to display ID'S

press 3 to delete the ID'S

press 4 to enter new job id's

3

ALL ID'S ARE DELETED!!

do you want to continue using the system?(y/n):n

\*/