Multiple Priority queue

```
# include<stdio.h>
# include<Stdlib.h>
# define N 3
int queue [3] [N];
int front [3]={0,0,0};
int rear [3]={-1, -1, -1};
int item, pr;
void main ()
 {
 intch;
 while (1)
 {
 printf ("Priority Queue \n");
 printf ("**** \n");
 printf ("\n\t1: PQ insert \n");
 printf ("\n\t2: PQ delete \n");
 printf ("\n\t3: PQ display \n");
 printf ("\n\t4: Exit \n");
 printf ("\n enter the choice \n");
 scanf ("%d", &ch);
 Switch (ch)
 {
   Case 1: Printf("\n enter the priority number\n
           scanf ("%d", &pr);
if (Pr>0 && Pr <4)
 pqinsert (PR- 1);
  else
printf ("\only 3 Priority exists 1 2 3 \n");
 break;
Case 2: Pqdelete ();
          break;
Case 3: display ();
          break;
Case 4: exit (0);
}
}
}
Pqinsert (int pr)
{
 if (rear[Pr]==N-1)
printf ("\n Queue over flow \n");
else
{
 Printf ("\n enter the item \n");
 Scanf ("%d", &item);
rear[Pr]++;
queue [Pr][rear [Pr]] = item;
}
```

```
 return;
 }
 Pqdelete()
{
int i;
for(i=0;i<3;i++)
 {
  if(rear[i]==front[i]-1)
  printf("\queue empty \n");
  else
  {
  Printf("deleted item is %d of queue %d \n",
         queue[i][front[i]],i+1;
      front[i]++;
      return;
    }
  }
}
display()
{
 int i,j;
for(i=0;i<3;i++)
{
 if(rear[i]==front[i]-1)
printf (" \queue is empty %d \n", i+1);
else
{printf ("\nqueue.%d:", i+1);
   for (j = front[i]; j<=rear[i]; j++)
     printf ("%dt", queue[i][j]);
   }
 }
 return;
}.
```