

```

//0-1 Knapsack problem using Greedy Algorithm
/*Output:
400*/
public class Knapsack{
    static int max(int a, int b)
    {
        return (a>b)?a:b;
    }
    int knapsack(int W, int wt[], int val[], int n)
    {
        int i,w;
        int K[][]=new int[n+1][W+1];
        for(i=0;i<=n;i++)
        {
            for (w=0;w<=W;w++)
            {
                if(i==0||w==0)
                    K[i][w]=0;
                else if(wt[i-1]<=w)
                    K[i][w]=max(val[i-1]+K[i-1][w-
wt[i-1]],K[i-1][w]);
                else
                    K[i][w]=K[i-1][w];
            }
        }
        return K[n][W];
    }
    public static void main(String[] args)
    {
        Knapsack ob=new Knapsack();
        int w[]={40,10,20,24}; //Weight
        int v[]={280,100,120,120}; //Value
        int W=65;//Capacity
        int size=w.length;
        System.out.print(ob.knapsack(W,w,v,size));
    }
}

```