#include <iostream>  
#include <algorithm>  
#include <set>  
using namespace std;  
class edge{  
public:  
 int from;  
 int to;  
 int dis;  
 edge(int f,int t,int d):from(f),to(t),dis(d){};  
};  
bool cmp(const edge &a,const edge &b){  
 return a.dis<b.dis;  
}  
int main(){  
 int n,m;  
 cin>>n>>m;  
 vector<edge> G;  
 for(int i=0;i<m;i++){  
 int a,b,c;  
 cin>>a>>b>>c;  
 G.push\_back({a-1,b-1,c}); //序号均-1存储，便于查找  
 }  
 sort(G.begin(),G.end(),cmp);  
 vector<edge> out;  
 int all=0;  
 vector<set<int>> S;  
 S.resize(n);  
 for(int i=0;i<n;i++)  
 S[i].insert(i);  
 int now=0;  
 while(out.size()!=n-1){  
 if(S[G[now].from]!=S[G[now].to]){  
 S[G[now].from].insert(G[now].to);  
 S[G[now].to]=S[G[now].from];  
 all+=G[now].dis;  
 out.push\_back(G[now]);  
 now++;  
 }  
 else{  
 now++;  
 }  
 }  
 cout<<all<<endl;  
 for(int i=0;i<out.size();i++){  
 cout<<out[i].from+1<<" "<<out[i].to+1<<" "<<out[i].dis<<endl;  
 }  
 return 0;  
}



