#include <iostream>  
#include <queue>  
#include <vector>  
using namespace std;  
class edge{  
public:  
 int dis;  
 int to;  
 edge(int t,int d):to(t),dis(d){};  
};  
class ans{  
public:  
 int num;  
 int dis;  
 int last;  
 ans(int n,int d,int l):num(n),dis(d),last(l){};  
 bool operator > (const ans &x) const{  
 return this->dis > x.dis;  
 }  
};  
int main(){  
 int n,m;  
 cin>>n>>m;  
 vector<vector<edge>> G;  
 G.resize(n);  
 vector<bool> vis;  
 vis.resize(n);  
 int all=0;  
 for(int i=0;i<n;i++)  
 vis[i]=0;  
 vector<ans> out;  
 priority\_queue<ans,vector<ans>,greater<ans>> pq;  
 for(int i=0;i<m;i++){  
 int a,b,c;  
 cin>>a>>b>>c;  
 G[a-1].push\_back({b-1,c}); //所有的顶点都-1存储 方便查找  
 G[b-1].push\_back({a-1,c});  
 }  
 //开始  
 vis[0]=1;  
 pq.push({0,0,0});  
 bool first=1;  
 while(!pq.empty()){  
 ans u = pq.top();  
 pq.pop();  
 if(vis[u.num]==1&&first!=1) continue;  
 if(first==1)first=0;  
 vis[u.num]=1;  
 out.push\_back({u.num,u.dis,u.last});  
 all+=u.dis;  
 for(int i=0;i<G[u.num].size();i++){  
 pq.push({G[u.num][i].to,G[u.num][i].dis,u.num});  
 }  
 }  
 cout<<all<<endl;  
 for(int i=1;i<out.size();i++)  
 cout<<out[i].last+1<<" "<<out[i].num+1<<" "<<out[i].dis<<endl;  
 return 0;  
}



