```cpp
// Test of ParticleType, ResonanceType and Particle classes - Luca Morelli 2021

#include "ParticleType.hpp"
#include "ResonanceType.hpp"
#include "Particle.hpp"

#include <vector>
#include <iostream>

void newLine(){
    std::cout<<'\n';
}

int main(){
    ParticleType p1{"Pione",100.1,1};
    p1.print();
    newLine();
    ResonanceType p2{"Muone",20.3,-3,12.3};
    p2.print();
    newLine();
    std::vector<ParticleType*> pVector{new ParticleType{p1}};
    pVector.push_back(new ResonanceType{p2});
    pVector[0]->print();
    newLine();
    pVector[1]->print();
    newLine();
    Particle::printParticleTypes();
    Particle::addParticleType("Pione",100,0);
    Particle pione{"Pione",10,-20,0};
    Particle::addParticleType("Caone",110,0,10);
    Particle caone{"Caone",100,3,-.70};
    pione.setParticle("Caone");
    std::cout<<"I:"<<caone.getIndex()<<'\n';
    pione.setParticle("Pione");
    Particle::printParticleTypes();
    pione.printDetails();
    newLine();
    std::cout<<"Energy:"<<pione.getEnergy()<<'\n';
    caone.printDetails();
    newLine();
    std::cout<<"Energy:"<<caone.getEnergy()<<'\n'<<"Invariant Mass:"
<<caone.invMass(pione)<<'\n';
    pione.setP(300,200,100);
    pione.printDetails();
    newLine();
    std::cout<<pione<<'\n'<<caone<<'\n'<<p1<<'\n';
    return 0;
}
```