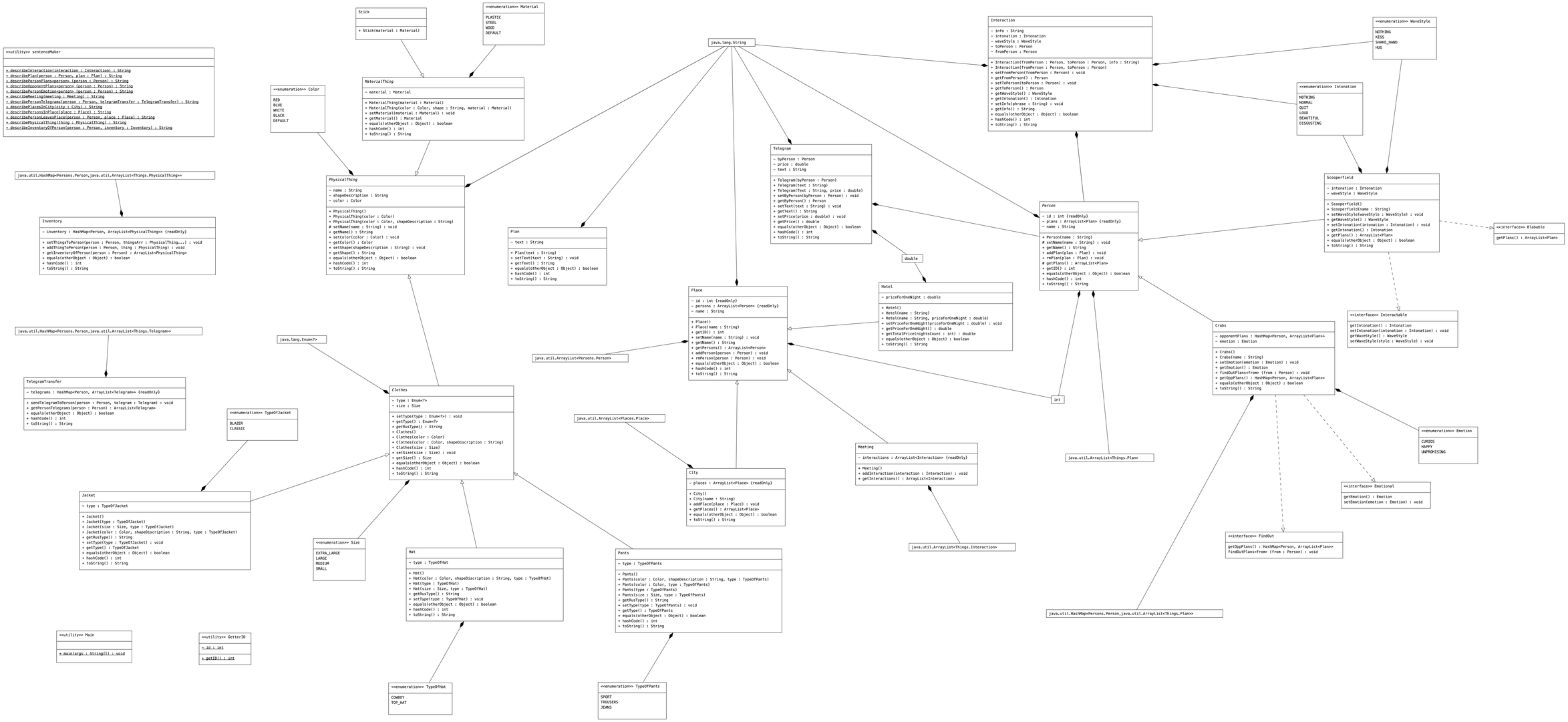
Stick
+ Stick(material : Material)
<<enumeration>> Material
PLASTIC
STEEL
WOOD
DEFAULT
<<utility>> sentenceMaker
+ describeInteraction(interaction : Interaction) : String
+ describePlan(person : Person, plan : Plan) : String
+ describePersonPlans(person : Person) : String
+ describeOpponentPlans(person : Person) : String
+ describePersonEmotion(person : Person) : String
+ describeMeeting(meeting : Meeting) : String
+ describePersonTelegrams(person : Person, telegramTransfer : TelegramTransfer) : String
+ describePlacesInCity(city : City) : String
+ describePersonsInPlace(place : Place) : String
+ describePersonsLeavesPlace(person : Person, place : Place) : String
+ describePhysicalThing(thing : PhysicalThing) : String
+ describeInventoryOfPerson(person : Person, inventory : Inventory) : String
<<enumeration>> Color
RED
BLUE
WHITE
BLACK
DEFAULT
MaterialThing
- material : Material
+ MaterialThing(material : Material)
+ MaterialThing(color : Color, shape : String, material : Material)
+ setMaterial(material : Material) : void
+ getMaterial() : Material
+ equals(otherObject : Object) : boolean
+ hashCode() : int
+ toString() : String
java.lang.String
Interaction
- info : String
- intonation : Intonation
- waveStyle : WaveStyle
- toPerson : Person
- fromPerson : Person
+ Interaction(fromPerson : Person, toPerson : Person, info : String)
+ Interaction(fromPerson : Person, toPerson : Person)
+ setFromPerson(fromPerson : Person) : void
+ getFromPerson() : Person
+ setToPerson(toPerson : Person) : void
+ getToPerson() : Person
+ getWaveStyle() : WaveStyle
+ getIntonation() : Intonation
+ setInfo(phrase : String) : void
+ getInfo() : String
+ equals(otherObject : Object) : boolean
+ hashCode() : int
+ toString() : String
<<enumeration>> WaveStyle
NOTHING
KISS
SHAKE_HAND
HUG
<<enumeration>> Intonation
NOTHING
NORMAL
QUIT
LOUD
BEAUTIFUL
DISGUSTING
java.util.HashMap<Persons.Person,java.util.ArrayList<Things.PhysicalThing>>
Inventory
- inventory : HashMap<Person, ArrayList<PhysicalThing>> {readOnly}
+ setThingsToPerson(person : Person, thingsArr : PhysicalThing...) : void
+ addThingToPerson(person : Person, thing : PhysicalThing) : void
+ getInventoryOfPerson(person : Person) : ArrayList<PhysicalThing>
+ equals(otherObject : Object) : boolean
+ hashCode() : int
+ toString() : String
PhysicalThing
- name : String
- shapeDescription : String
- color : Color
+ PhysicalThing()
+ PhysicalThing(color : Color)
+ PhysicalThing(color : Color, shapeDescription : String)
# setName(name : String) : void
+ getName() : String
+ setColor(color : Color) : void
+ getColor() : Color
+ setShape(shapeDescription : String) : void
+ getShape() : String
+ equals(otherObject : Object) : boolean
+ hashCode() : int
+ toString() : String
Plan
- text : String
+ Plan(text : String)
+ setText(text : String) : void
+ getText() : String
+ equals(otherObject : Object) : boolean
+ hashCode() : int
+ toString() : String
Telegram
- byPerson : Person
- price : double
- text : String
+ Telegram(byPerson : Person)
+ Telegram(text : String)
+ Telegram(Text : String, price : double)
+ setByPerson(byPerson : Person) : void
+ getByPerson() : Person
+ setText(text : String) : void
+ getText() : String
+ setPrice(price : double) : void
+ getPrice() : double
+ equals(otherObject : Object) : boolean
+ hashCode() : int
+ toString() : String
Person
- id : int {readOnly}
- plans : ArrayList<Plan> {readOnly}
- name : String
+ Person(name : String)
# setName(name : String) : void
+ getName() : String
+ addPlan(plan : Plan) : void
+ rmPlan(plan : Plan) : void
# getPlans() : ArrayList<Plan>
+ getID() : int
+ equals(otherObject : Object) : boolean
+ hashCode() : int
+ toString() : String
Scooperfield
- intonation : Intonation
- waveStyle : WaveStyle
+ Scooperfield()
+ Scooperfield(name : String)
+ setWaveStyle(waveStyle : WaveStyle) : void
+ getWaveStyle() : WaveStyle
+ setIntonation(intonation : Intonation) : void
+ getIntonation() : Intonation
+ getPlans() : ArrayList<Plan>
+ equals(otherObject : Object) : boolean
+ toString() : String
<<interface>> Blabable
getPlans() : ArrayList<Plan>
double
Hotel
- priceForOneNight : double
+ Hotel()
+ Hotel(name : String)
+ Hotel(name : String, priceForOneNight : double)
+ setPriceForOneNight(priceForOneNight : double) : void
+ getPriceForOneNight() : double
+ getTotalPrice(nightsCount : int) : double
+ equals(otherObject : Object) : boolean
+ toString() : String
Place
- id : int {readOnly}
- persons : ArrayList<Person> {readOnly}
- name : String
+ Place()
+ Place(name : String)
+ getID() : int
+ setName(name : String) : void
+ getName() : String
+ getPersons() : ArrayList<Person>
+ addPerson(person : Person) : void
+ rmPerson(person : Person) : void
+ equals(otherObject : Object) : boolean
+ hashCode() : int
+ toString() : String
<<interface>> Interactable
getIntonation() : Intonation
setIntonation(intonation : Intonation) : void
getWaveStyle() : WaveStyle
setWaveStyle(style : WaveStyle) : void
Crabs
- opponentPlans : HashMap<Person, ArrayList<Plan>>
- emotion : Emotion
+ Crabs()
+ Crabs(name : String)
+ setEmotion(emotion : Emotion) : void
+ getEmotion() : Emotion
+ findOutPlans<from> (from : Person) : void
+ getOppPlans() : HashMap<Person, ArrayList<Plan>>
+ equals(otherObject : Object) : boolean
+ toString() : String
java.util.HashMap<Persons.Person,java.util.ArrayList<Things.Telegram>>
java.lang.Enum<?>
java.util.ArrayList<Persons.Person>
int
TelegramTransfer
- telegrams : HashMap<Person, ArrayList<Telegram>> {readOnly}
+ sendTelegramToPerson(person : Person, telegram : Telegram) : void
+ getPersonTelegrams(person : Person) : ArrayList<Telegram>
+ equals(otherObject : Object) : boolean
+ hashCode() : int
+ toString() : String
Clothes
- type : Enum<?>
- size : Size
+ setType(type : Enum<?>) : void
+ getType() : Enum<?>
+ getRusType() : String
+ Clothes()
+ Clothes(color : Color)
+ Clothes(color : Color, shapeDiscription : String)
+ Clothes(size : Size)
+ setSize(size : Size) : void
+ getSize() : Size
+ equals(otherObject : Object) : boolean
+ hashCode() : int
+ toString() : String
<<enumeration>> TypeOfJacket
BLAZER
CLASSIC
java.util.ArrayList<Places.Place>
Meeting
- interactions : ArrayList<Interaction> {readOnly}
+ Meeting()
+ addInteraction(interaction : Interaction) : void
+ getInteractions() : ArrayList<Interaction>
City
- places : ArrayList<Place> {readOnly}
+ City()
+ City(name : String)
+ addPlace(place : Place) : void
+ getPlaces() : ArrayList<Place>
+ equals(otherObject : Object) : boolean
+ toString() : String
java.util.ArrayList<Things.Plan>
<<enumeration>> Emotion
CURIOS
HAPPY
UNPROMISING
<<interface>> Emotional
getEmotion() : Emotion
setEmotion(emotion : Emotion) : void
Jacket
~ type : TypeOfJacket
+ Jacket()
+ Jacket(type : TypeOfJacket)
+ Jacket(size : Size, type : TypeOfJacket)
+ Jacket(color : Color, shapeDiscription : String, type : TypeOfJacket)
+ getRusType() : String
+ setType(type : TypeOfJacket) : void
+ getType() : TypeOfJacket
+ equals(otherObject : Object) : boolean
+ hashCode() : int
+ toString() : String
<<enumeration>> Size
EXTRA_LARGE
LARGE
MEDIUM
SMALL
Hat
~ type : TypeOfHat
+ Hat()
+ Hat(color : Color, shapeDiscription : String, type : TypeOfHat)
+ Hat(type : TypeOfHat)
+ Hat(size : Size, type : TypeOfHat)
+ getRusType() : String
+ setType(type : TypeOfHat) : void
+ equals(otherObject : Object) : boolean
+ hashCode() : int
+ toString() : String
Pants
~ type : TypeOfPants
+ Pants()
+ Pants(color : Color, shapeDescription : String, type : TypeOfPants)
+ Pants(color : Color, type : TypeOfPants)
+ Pants(type : TypeOfPants)
+ Pants(size : Size, type : TypeOfPants)
+ getRusType() : String
+ setType(type : TypeOfPants) : void
+ getType() : TypeOfPants
+ equals(otherObject : Object) : boolean
+ hashCode() : int
+ toString() : String
<<interface>> FindOut
getOppPlans() : HashMap<Person, ArrayList<Plan>>
findOutPlans<from> (from : Person) : void
java.util.ArrayList<Things.Interaction>
java.util.HashMap<Persons.Person,java.util.ArrayList<Things.Plan>>
<<utility>> Main
+ main(args : String[]) : void
<<utility>> GetterID
- id : int
+ getID() : int
<<enumeration>> TypeOfHat
COWBOY
TOP_HAT
<<enumeration>> TypeOfPants
SPORT
TROUSERS
JEANS