# Number Prettifier Design Document

## Contents

## Overview

Design a simple application which will format numbers based on the user input and requirements.

The program purpose is to make the numbers readable to human viewer.

## Requirements

1. Numbers that contain less than 7 digits (not including decimal parts) should not be changed for example the number “654123” will not be changed and the program will return the same number in the same format.
2. If decimal part exists then the program will return only the first decimal digit. For example the number “543.3213123” would be transformed to “543.3”
3. Numbers that have between 7 and 10 digits are considered MEGA numbers and would be prettified like the following example: “1234567” would be transformed to 1.2M.
4. Numbers that have between 10 and 13 digits are considered GIGA numbers and would be prettified like the following example: “1234567890” would be transformed to 1.2B.
5. Numbers that have more than 13 are considered TERA numbers and would be prettified like the following example: “1234567890123” would be transformed to 1.2T.
6. Program should handle erroneous user input.

## Assumptions

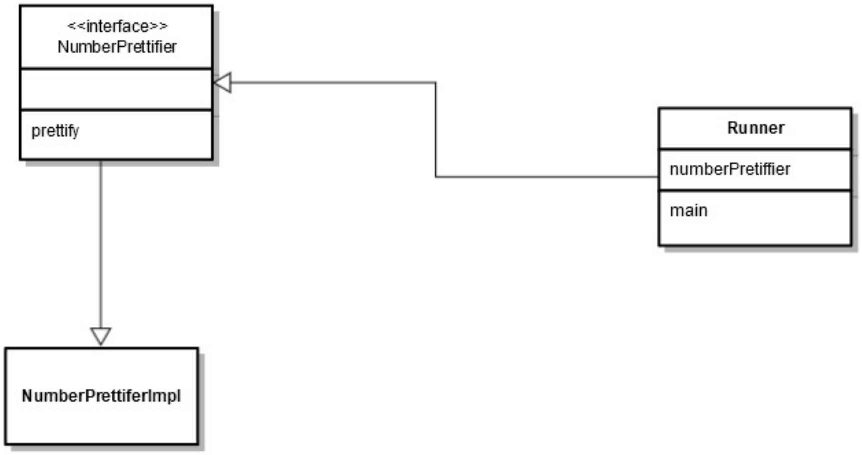
1. Numbers will be rounded.
2. The program will prettify numbers in the millions, billions and trillions only.
3. We will not include thousand separator.

## Technical Design

Since the formatting requirements are probably flexible and might change over time or even language and locale, first create and interface that specify the “prettify” method

This way should future requirements arrive it would be easier to change the implementing class

Please regard the following UML diagram for more details



## The Runner Class

The Runner class is the entry point to the application, and its responsibilities involve instantiating the interface using the implementing class (only one implementing class possible at the moment) and checking for user input.

This class exists inside the package com.uri.mobi.runner

And contains a loop that listen to user input (using Scanner) and making basic validation that the input is indeed a valid number.

In case it is the method invokes the method prettify of the interface.

## The NumberPrettifier

```
public interface NumberPrettifier
```

This interface will exist in the package com.uri.mobi.service and will define the method prettify

Using the following signature:

```
public String prettify(float numberToPrettify);
```

Also it will define the static final longs needed to define the formatting

```
public static final long MEGA = 1000000;
public static final long GIGA = 1000000000;
public static final long TERA = 1000000000000l;
```

## The Implementing class

Inside the same package create the class

```
NumberPrettifierImpl
```

This class will implement the logic needed to do the actual formatting

It will contain some private methods to evaluate the number such as

```
private boolean isLessThanMega(BigDecimal number)

private boolean isMoreThanTeraNumber(BigDecimal number

private boolean isMegaNumber(BigDecimal number)
```

And will do the actual formatting accordingly.