YAML and JSON Hands-On Exercises

For each exercise, create a YAML document with required data, and test its validity using **yamllint** or its online equivalent (http://www.yamllint.com/).

Optional: create corresponding JSON data and validate it with online JSON validator (for example, https://jsonlint.com/).

1. Create a dictionary[1] containing **hostname** (a string), **loopback** (an IP address), **maxint** (number) and **banner** (6 lines, preserve the newlines).
2. Create a dictionary containing **hostname** (a string), **loopback** (an IP address), and **interfaces** (a list of strings).
3. Create a list[2] of hostnames (each hostname is a FQDN string)
4. Create list of **username** (string) / **password** (string) pairs.
5. Create a list of users having:
   - Mandatory[3]: **username** and **password**
   - Optional[4]: **privilege** (number) and **autocommand** (string) attributes
6. Create a list of BGP neighbors having:
   - Mandatory: **address** (IP address)**, description** (string) and **asn**
   - Optional: **state** (**present, absent** or **down**) and **policy** (route map name)
7. Create a list of BGP neighbors as above, add optional attribute **prefixes** (list of IP prefixes). Use inline list syntax for the **prefixes** attribute.
8. Create a BGP router description containing:
   - **hostname** (string)**, loopback** (IP address)**, asn** (number) and **banner** (6 lines, preserve the newlines)
   - **interfaces**: list of **name / address** pairs, where every **address** is an IP address using CIDR prefix notation (for example, 192.168.1.1/24). Use inline dictionary syntax for the elements of this list.
   - **neighbors**: list of BGP neighbors (see above)

---

1 We use Python terminology. You could call them mappings, objects or hashes.

2 More Python terminology. Feel free to call them arrays or sequences.

3 Each element should have these values.

4 Only some elements should have these values.