

```
#!/usr/bin/env python3
# title      : My_resume.py
# description : Useful information about my career
# author     : Wojtek Zygodlo
# email:     : wzygodlo@outlook.com
# phone:     : +1 778 751 3750
# version    : 1.0
# python_version : 3.6
```

```
import education
import work_experience as work
import skills
import hobby
import time
```

```
class Resume:
    def __init__(level="experienced"):
        super()

    def bio(self):
        info = '''Overall 10 years of enterprise level UNIX administration experience with strong Linux
        (RedHat, SLES, Ubuntu, Archlinux is the best), Scripting (BASH, Python).
        Deep understanding on CISCO networks and hardware, quick problem resolution skills.
        Around 2,5 years experience in Mobile Networks (UMTS).
        4 years in Broadcasting industry (satellite/cable IPTV solutions)
        4 years in DevOps environment'''
        print(info)

    def skills(self):
        skill = input()
        if skill == "Linux":
            print('''RHEL5/6/7, Debian, Ubuntu, SLES 11/12, Archlinux, and many more''')

        elif skill == "Network Skills":
            print('''Switching (VLAN, Trunking, Link Aggregation), Wireless
            Routing (RIPv2, OSPF, EIGRP),
            Load balancing (HAProxy),
            FibreChannel,
            TCP/IP,
            Network monitoring and analysing (Wireshark, nmap, Nagios),
            Server setup (Apache, nginx, SSH, FTP, DHCP, NTP, SMB, NFS, OpenLDAP, VPN, AFP, SSL)''')

        elif skill == "Programming":
            print("Shell/BASH", "Python")

        elif skill == "Database":
            print('''Oracle 10/11g DBA"
            MySQL DBA,
            MongoDB (query level),
            SQL/SQLPlus''')

        elif skill == "Cloud Solutions":
            print("AWS (EC2, VPC, CloudFormation, Lambda, Route53....)"
            "Openstack"
            "Vagrant"
            "Docker"
            )

        elif skill == "DevOps Tools":
            print(''' Teamcity/Jenkins/Travis CI
            Flyways,
            GitHub
            Jfrog Artifactory,
            ELK / NewRelic / Nagios,
            SonarQube
            ''')

    def work_history(self):
        work = input("Work History: ")
        start_time = time.strftime("%m/%Y")
        end_time = time.strftime("%m/%Y")

        if start_time == "08/2017" and end_time == "current":
            work_history = {"PayPal/Hyperwallet Systems Inc": "DevOps Engineer / Linux Senior Admin",
                            "Location": "Vancouver, Canada",
                            "Duties": ["A lot"]}
            print(work_history)
```

```

elif start_time == "05/2013" and end_time == "07/2017":
    work_history = {"Cisco Systems LTD": "Systems Engineer",
                    "Location": "Southampton, UK",
                    "Duties": ['''
                        - Managing and monitoring all installed systems and infrastructure
                          to ensure the highest level of availability. (Nagios/ELK/Grafana).
                        - Installing, configuring and testing operating systems, application software.
                        - Managing security, upgrade and deployment projects on-site.
                        - Built and provided high-level E2E, troubleshooting on Windows, Linux Systems.
                        - Overseeing development and maintenance of customers computer systems
                          in Broadcasting industry,
                        - Maintain 24/7 availability for responsible systems. (On-call rotation).
                        - Providing CI based on Jenkins solution.
                        - Monitoring and testing application performance to identify potential bottlenecks,
                          developing solutions, and collaborate with developers
                          on solution implementation.
                        - Writing and maintaining custom scripts to increase system efficiency
                          and performance time.
                        - Providing 2nd and 3rd level technical support and troubleshooting
                          to internal and external clients.
                    ''']]
    print(work_history)

elif start_time == "02/2011" and end_time == "05/2013":
    work_history = {"Nokia Siemens Network": "Technical Support Engineer",
                    "Location": "Wroclaw, Poland",
                    "Duties": ['''
                        - Testing (E2E, IOT),
                        - Trailing activities,
                        - Providing 24/7 support (on-call rotation),
                        - Design and implementation of test environment
                          based on Linux RedHat servers.,
                        - Implementing VLAN solution at the customer side
                          for Femto/Small Cells network.,
                        - Features verification, testing new features in new SW release.,
                        - Remote and onsite support for customers all over the world,
                        - Providing internal training for customer teams regarding NSN Femto Solution,
                        - Managing and maintenance experience in Core Networks (MSC, SGSN)
                          during Femto integration at customer sites,
                        - Tracking problems and solutions based on Resolve BMC Remedy,
                          JIRA, Service Desk."
                    ''']]
    print(work_history)

elif start_time == "10/2007" and end_time == "02/2011":
    work_history = {"TECE": "Network Administrator",
                    "Location": "Strzelin, Poland",
                    "Duties": ['''
                        - Installation, upgrading and maintaining Servers Windows Server 2003 R2,
                        - Managing Web server (Apache, IIS),
                        - Linux server administration based on Debian
                          (DHCP, Proxy server, Samba, IPtables)
                          Novell SLES (Backup email server based on eDirecotry)
                          Ubuntu with KVM for virtualisation.,
                        - Cisco device management (Switches, Routers),
                        - Configuration routeing table,
                        - Maintaining VPN connection between TECE headquarter and branch offices,
                        - Creating websites (PHP, CSS, HTML),
                        - Database administration of MySQL, MSSQL,
                        - Helpdesk for 60 peoples in the TECE Poland office.
                    ''']]
    print(work_history)

elif start_time == "11/2005" and end_time == "10/2007":
    work_history = {"PHOUA SP z o.o.": "Network Administrator",
                    "Locaton": "Wroclaw, Poland",
                    "Duties": ['''
                        - Managing network infrastructure
                        - Implementation WLAN access on Cisco devices
                        - Maintaining Windows systems
                        - Helpdesk
                    ''']]
    print(work_history)

elif start_time == "01/2005" and end_time == "09/2005":
    work_history = {"Minimarket Max": "Help Desk",

```

```

        "Location": "Strzelin, Poland",
        "Duties": "Programming fiscal cash registers."}
    print(work_history)

def education(self):
    edu = input("Education: ")
    start_time = time.strftime("%m/%Y")

    if start_time == "01/2010":
        education = {"Bachelor of Science": "Computer Systems and Networks",
                     "School Name": "Wroclaw High School of applied computing (WWSIS)",
                     "School Location": "Wroclaw, Poland"}

    elif start_time == "06/2006":
        education = {"Post-secondary school": "IT",
                     "School Name": "Secondary Study IT 'WIEDZA'",
                     "School Location": "Wroclaw, Poland"}

    elif start_time == "06/2004":
        education = {"Post-secondary school": "IT",
                     "School Name": "EZN Electronic Departments",
                     "School Location": "Wroclaw, Poland"}

def hobby(self):
    while True:
        hobby = input("Do you have a hobby: ")
        if str.lower(hobby) == "y":
            print("Photography, Hiking, Football, Cyber Security, Marial Arts")

```