```
<?xml version="1.0" encoding="UTF-8"?>
<indexing>
 <paragraph index="9" node_type="writer">Xander Harris</paragraph>
 <paragraph index="10" node_type="writer">Principal DevOps Engineer | Staff
Software Engineer | Cloud Architect</paragraph>
 <object index="11" name="TextSection" object_type="section"/>
 <paragraph index="12" node_type="writer" parent_index="11">Contact</paragraph>
 <paragraph index="13" node_type="writer" parent_index="11">+1(213) 285-7203 |
xandertheharris@gmail.com | LinkedIn | GitHub | HackerRank</paragraph>
 <paragraph index="14" node_type="writer" parent_index="11">Summary</paragraph>
 <paragraph index="15" node_type="writer" parent_index="11">A straight shooter with
upper management written all over him.</paragraph>
 <paragraph index="16" node_type="writer" parent_index="11">Skills</paragraph>
 <object index="18" name="Section1" object_type="section"/>
 <paragraph index="19" node_type="writer" parent_index="18">Automation</paragraph>
 <paragraph index="20" node_type="writer" parent_index="18">OSS</paragraph>
 <paragraph index="21" node_type="writer" parent_index="18">Terraform</paragraph>
 <paragraph index="22" node_type="writer" parent_index="18">Ansible</paragraph>
 <paragraph index="23" node_type="writer" parent_index="18">Jenkins</paragraph>
 <paragraph index="24" node_type="writer" parent_index="18">Google
Cloud</paragraph>
 <paragraph index="25" node_type="writer" parent_index="18">GDM</paragraph>
 <paragraph index="26" node_type="writer" parent_index="18">AWS</paragraph>
 <paragraph index="27" node_type="writer" parent_index="18">Cloud
Formation</paragraph>
 <paragraph index="28" node_type="writer" parent_index="18">Monitoring</paragraph>
 <paragraph index="29" node_type="writer" parent_index="18">OSS</paragraph>
 <paragraph index="30" node_type="writer" parent_index="18">Prometheus</paragraph>
 <paragraph index="31" node_type="writer" parent_index="18">Grafana</paragraph>
 <paragraph index="32" node_type="writer" parent_index="18">Thanos</paragraph>
 <paragraph index="33" node_type="writer"
parent_index="18">AlertManager</paragraph>
 <paragraph index="34" node_type="writer" parent_index="18">Loki</paragraph>
 <paragraph index="35" node_type="writer" parent_index="18">MaaS</paragraph>
 <paragraph index="36" node_type="writer" parent_index="18">NewRelic</paragraph>
 <paragraph index="37" node_type="writer" parent_index="18">DataDog</paragraph>
 <paragraph index="38" node_type="writer" parent_index="18">StackDriver</paragraph>
 <paragraph index="39" node_type="writer" parent_index="18">ServiceNow</paragraph>
 <paragraph index="40" node_type="writer" parent_index="18">Networking</paragraph>
 <paragraph index="41" node_type="writer" parent_index="18">GCP/AWS</paragraph>
 <paragraph index="42" node_type="writer" parent_index="18">VPCs</paragraph>
 <paragraph index="43" node_type="writer" parent_index="18">Subnets</paragraph>
 <paragraph index="44" node_type="writer" parent_index="18">Security
Groups</paragraph>
 <paragraph index="45" node_type="writer" parent_index="18">Firewalls</paragraph>
 <paragraph index="46" node_type="writer" parent_index="18">General</paragraph>
 <paragraph index="47" node_type="writer" parent_index="18">TCP/IP</paragraph>
 <paragraph index="48" node_type="writer" parent_index="18">UDP</paragraph>
 <paragraph index="49" node_type="writer" parent_index="18">E2E
Encryption</paragraph>
 <paragraph index="51" node_type="writer" parent_index="18">Databases</paragraph>
 <paragraph index="52" node_type="writer" parent_index="18">Relational
Databases</paragraph>
 <paragraph index="53" node_type="writer" parent_index="18">PostgreSQL</paragraph>
 <paragraph index="54" node_type="writer" parent_index="18">MariaDB</paragraph>
 <paragraph index="55" node_type="writer" parent_index="18">MySQL</paragraph>
 <paragraph index="56" node_type="writer" parent_index="18">MSSQL</paragraph>
 <paragraph index="57" node_type="writer" parent_index="18">NoSQL/Big
Data</paragraph>
```

<paragraph index="58" node_type="writer" parent_index="18">MongoDB</paragraph>
<paragraph index="59" node_type="writer" parent_index="18">DynamoDB</paragraph>
<paragraph index="60" node_type="writer" parent_index="18">Google Big Table</paragraph>
<paragraph index="61" node_type="writer" parent_index="18">Google Big Query</paragraph>
<paragraph index="62" node_type="writer" parent_index="18">Languages</paragraph>
<paragraph index="63" node_type="writer" parent_index="18">Programming</paragraph>
<paragraph index="64" node_type="writer" parent_index="18">Python</paragraph>
<paragraph index="65" node_type="writer" parent_index="18">Go (Golang)</paragraph>
<paragraph index="66" node_type="writer" parent_index="18">C</paragraph>
<paragraph index="67" node_type="writer" parent_index="18">Java</paragraph>
<paragraph index="68" node_type="writer" parent_index="18">JavaScript</paragraph>
<paragraph index="69" node_type="writer" parent_index="18">Scripting</paragraph>
<paragraph index="70" node_type="writer" parent_index="18">Shell</paragraph>
<paragraph index="71" node_type="writer" parent_index="18">BASH</paragraph>
<paragraph index="72" node_type="writer" parent_index="18">PowerShell</paragraph>
<paragraph index="73" node_type="writer" parent_index="18">Nix</paragraph>
<paragraph index="74" node_type="writer" parent_index="18">Markup</paragraph>
<paragraph index="75" node_type="writer" parent_index="18">Markdown</paragraph>
<paragraph index="76" node_type="writer" parent_index="18">XML</paragraph>
<paragraph index="77" node_type="writer" parent_index="18">HTML</paragraph>
<paragraph index="78" node_type="writer" parent_index="18">YAML</paragraph>
<paragraph index="80" node_type="writer" parent_index="18">Orchestration</paragraph>
<paragraph index="81" node_type="writer" parent_index="18">Platforms</paragraph>
<paragraph index="82" node_type="writer" parent_index="18">Bare-metal Kubernetes</paragraph>
<paragraph index="83" node_type="writer" parent_index="18">Docker Enterprise</paragraph>
<paragraph index="84" node_type="writer" parent_index="18">Container D</paragraph>
<paragraph index="85" node_type="writer" parent_index="18">CRI-O</paragraph>
<paragraph index="86" node_type="writer" parent_index="18">GCP</paragraph>
<paragraph index="87" node_type="writer" parent_index="18">GKE (Google Kubernetes Engine)</paragraph>
<paragraph index="88" node_type="writer" parent_index="18">GCE (Google Compute Engine)</paragraph>
<paragraph index="89" node_type="writer" parent_index="18">AWS</paragraph>
<paragraph index="90" node_type="writer" parent_index="18">EKS (Elastic Kubernetes Service)</paragraph>
<paragraph index="91" node_type="writer" parent_index="18">ECS (Elastic Container Service)</paragraph>
<paragraph index="92" node_type="writer" parent_index="18">EC2 (Elastic Compute Cloud)</paragraph>
<paragraph index="93" node_type="writer" parent_index="18">CI/CD</paragraph>
<paragraph index="94" node_type="writer" parent_index="18">Build/Delivery</paragraph>
<paragraph index="95" node_type="writer" parent_index="18">Jenkins</paragraph>
<paragraph index="96" node_type="writer" parent_index="18">GitLab CI</paragraph>
<paragraph index="97" node_type="writer" parent_index="18">CircleCI</paragraph>
<paragraph index="98" node_type="writer" parent_index="18">ArgoCD</paragraph>
<paragraph index="99" node_type="writer" parent_index="18">Travis CI</paragraph>
<paragraph index="100" node_type="writer" parent_index="18">Quality</paragraph>
<paragraph index="101" node_type="writer" parent_index="18">SonarQube</paragraph>
<paragraph index="102" node_type="writer" parent_index="18">Code-Climate</paragraph>
<paragraph index="103" node_type="writer" parent_index="18">CodeCov</paragraph>
<paragraph index="104" node_type="writer" parent_index="18">Software Development Life Cycle</paragraph>

<paragraph index="105" node_type="writer" parent_index="18">Agile</paragraph>
<paragraph index="106" node_type="writer" parent_index="18">SCRUM</paragraph>
<paragraph index="107" node_type="writer" parent_index="18">Branching Models</paragraph>
<paragraph index="108" node_type="writer" parent_index="18">Waterfall</paragraph>
<paragraph index="109" node_type="writer" parent_index="18">Semantic Versioning</paragraph>
<paragraph index="110" node_type="writer" parent_index="18">Release Cycles</paragraph>
<paragraph index="111" node_type="writer" parent_index="18">Commit Cycles</paragraph>
<object index="113" name="Section2" object_type="section"/>
<paragraph index="114" node_type="writer" parent_index="113">Experience</paragraph>
<paragraph index="115" node_type="writer" parent_index="113">Platonic, Inc - Senior DevOps Engineer (10-2024 - 04 - 2024)</paragraph>
<paragraph index="116" node_type="writer" parent_index="113">Designed and implemented scalable blockchain services.</paragraph>
<paragraph index="117" node_type="writer" parent_index="113">Used and improved Ansible playbooks used in the generation of Kubernetes manifests.</paragraph>
<paragraph index="118" node_type="writer" parent_index="113">Stood up Elastic Kubernetes Service clusters and Google Compute Engine instances with HCP Terraform, AWS Console, and GCP Console.</paragraph>
<paragraph index="119" node_type="writer" parent_index="113">Built and deployed containers to EKS clusters using ECR, GCE, and Docker.</paragraph>
<paragraph index="120" node_type="writer" parent_index="113">Deployed custom services written in Go and Haskell to EKS clusters using GCE, Ansible, shell scripting and Helm.</paragraph>
<paragraph index="121" node_type="writer" parent_index="113">Converted Kubernetes manifests as generated by Ansible and shell scripts into Helm charts to speed deployment of blockchain networks.</paragraph>
<paragraph index="122" node_type="writer" parent_index="113">Was converted to the use of nix for generating consistent development environments across systems, my life will never be the same.</paragraph>
<paragraph index="123" node_type="writer" parent_index="113">Tools: git, neovim, nix, GitLab CI, Docker, BASH, Go, GCP GKE, Google Compute Engine, AWS S3, AWS ECR, PostgreSQL, Terraform, Python</paragraph>
<paragraph index="125" node_type="writer" parent_index="113">SiriusXM, Inc - Staff Software Engineer (04-2024 - 06-2024)</paragraph>
<paragraph index="126" node_type="writer" parent_index="113">Software Engineering</paragraph>
<paragraph index="127" node_type="writer" parent_index="113">Automated documentation processes with existing tools, including GitLab CI/CD, Python, and GitLab Pages.</paragraph>
<paragraph index="128" node_type="writer" parent_index="113">Created Python programs to automate and simplify versioning processes while maintaining Semantic Versioning and PEP-440 compliance.</paragraph>
<paragraph index="129" node_type="writer" parent_index="113">Supported and advised several development teams on containerization and migration from on premises infrastructure to AWS deployments.</paragraph>
<paragraph index="130" node_type="writer" parent_index="113">Infrastructure as Code</paragraph>
<paragraph index="131" node_type="writer" parent_index="113">Wrote Terraform modules to facilitate transfer of infrastructure from existing Data Centers to AWS accounts in all major environments (dev, qa, staging, prod).</paragraph>
<paragraph index="132" node_type="writer" parent_index="113">Used existing tools to produce documentation for existing and new Terraform modules where none existed previously.</paragraph>
<paragraph index="133" node_type="writer" parent_index="113">Provided advice to

and received requirements from development teams on the most cost-effective use of cloud resources, including deployments to EKS, VPCs, S3, Cognito and Lambda.</paragraph>
 <paragraph index="134" node_type="writer" parent_index="113">Tools: vscode, vim, GitLab CI, Docker, BASH, Go, GCP GKE, Compute Engine, Cloud Storage, Networking, Cloud SQL, Terraform, Python, git</paragraph>
 <paragraph index="135" node_type="writer" parent_index="113">Wunderkind, Inc – DevOps Engineer / Software Engineer (09-2021 - 12-2023)</paragraph>
 <paragraph index="136" node_type="writer" parent_index="113">Automation, Integration, Delivery and Operations</paragraph>
 <paragraph index="137" node_type="writer" parent_index="113">Created and maintained Helm charts that were used to deploy custom and open-source software to ~36 GKE clusters.</paragraph>
 <paragraph index="138" node_type="writer" parent_index="113">Supported development teams with GitLab CI/CD pipeline templates and pipelines for continuous delivery. </paragraph>
 <paragraph index="139" node_type="writer" parent_index="113">Contributed to organization-wide metric standards to ensure low cardinality and baseline metrics provided to Prometheus.</paragraph>
 <paragraph index="140" node_type="writer" parent_index="113">Completed migration from DataDog to Prometheus and Grafana by removing all DataDog-sourced alerts from Grafana.</paragraph>
 <paragraph index="141" node_type="writer" parent_index="113">Implemented SonarQube static analysis for custom Go and Python projects.</paragraph>
 <paragraph index="142" node_type="writer" parent_index="113">Developed multi-stage Docker builds to reduce build time and create smaller deployment images.</paragraph>
 <paragraph index="143" node_type="writer" parent_index="113">Deployed opencost.io across all existing GKE clusters that enabled cost analysis via Prometheus and Grafana.</paragraph>
 <paragraph index="144" node_type="writer" parent_index="113">Demonstrated usage of the GCP Billing Console to R&D executives.</paragraph>
 <paragraph index="145" node_type="writer" parent_index="113">Infrastructure as Code</paragraph>
 <paragraph index="146" node_type="writer" parent_index="113">Created Terraform code to deploy or update GKE clusters and node pools to follow organization labeling standards.</paragraph>
 <paragraph index="147" node_type="writer" parent_index="113">Imported all existing GCP infrastructure into GitLab Terraform State to enable a transition from GDM.</paragraph>
 <paragraph index="148" node_type="writer" parent_index="113">Migrated infrastructure code from Google Deployment Manager to GitLab CI with Terraform.</paragraph>
 <paragraph index="149" node_type="writer" parent_index="113">Go (Golang) Development and Software Engineering</paragraph>
 <paragraph index="150" node_type="writer" parent_index="113">Implemented Go metrics for MySQL, enhancing monitoring with real-time database performance insights.</paragraph>
 <paragraph index="151" node_type="writer" parent_index="113">Developed automated tests in Go for API validation, ensuring high reliability and performance across system components.</paragraph>
 <paragraph index="152" node_type="writer" parent_index="113">[‘hip’, ’hip’] (array!) </paragraph>
 <paragraph index="153" node_type="writer" parent_index="113">Tools: vscode, vim, GitLab CI, Docker, BASH, Go, GCP GKE, Compute Engine, Cloud Storage, Networking, Cloud SQL, Terraform, Python, git</paragraph>
 <paragraph index="154" node_type="writer" parent_index="113">Databricks, Inc – Part-Time Documentation Platform Engineer (11-2020 – 07-2023)</paragraph>
 <paragraph index="155" node_type="writer" parent_index="113">Python Development

and Software Engineering</paragraph>
 <paragraph index="156" node_type="writer" parent_index="113">Automated Python API docs for Databricks Feature Store, streamlining documentation processes and enhancing team efficiency.</paragraph>
 <paragraph index="157" node_type="writer" parent_index="113">Enhanced PyTest coverage for Databricks docs Python codebase, boosting code reliability and maintainability.</paragraph>
 <paragraph index="158" node_type="writer" parent_index="113">Automation, Integration, Delivery and Operations</paragraph>
 <paragraph index="159" node_type="writer" parent_index="113">Documented build/test workflows for documentation sites, ensuring consistent, error-free updates and deployments.</paragraph>
 <paragraph index="160" node_type="writer" parent_index="113">Managed CircleCI pipelines for Databricks docs, ensuring seamless builds/deployments and integrating custom Sphinx extensions.</paragraph>
 <paragraph index="161" node_type="writer" parent_index="113">Supported technical writers with the use of Sphinx to generate Databricks docs, ensuring quality and consistency in documentation delivery.</paragraph>
 <paragraph index="162" node_type="writer" parent_index="113">Tools: vscode, vim, Python, Sphinx, Bazel, CircleCI, git, GitHub, pytest, bash, Docker, YAML, Markdown, rST</paragraph>
 <paragraph index="163" node_type="writer" parent_index="113">Fox Corp - Senior DevOps Engineer - (05-2020 - 07-2021)</paragraph>
 <paragraph index="164" node_type="writer" parent_index="113">Monitoring with NewRelic and DataDog</paragraph>
 <paragraph index="165" node_type="writer" parent_index="113">Compiled data on New Relic alerts caused by Media Cloud systems for tuning purposes.</paragraph>
 <paragraph index="166" node_type="writer" parent_index="113">Moved existing New Relic monitoring to DataDog for all Media Cloud services.</paragraph>
 <paragraph index="167" node_type="writer" parent_index="113">Automated deployment of DataDog agent configuration to Docker services with Jenkins and Ansible.
</paragraph>
 <paragraph index="168" node_type="writer" parent_index="113">Automation, Integration, Delivery and Operations</paragraph>
 <paragraph index="169" node_type="writer" parent_index="113">Improved stability of video and audio transcoding and quality-control production services running on EC2 instances using both Linux and Windows such that outage frequency was reduced from several per day to once or twice per month.</paragraph>
 <paragraph index="170" node_type="writer" parent_index="113">Participated in a 24/7 on-call rotation to provide incident response and issue resolution for all Media Cloud mission-critical production services.</paragraph>
 <paragraph index="171" node_type="writer" parent_index="113">Wrote Jenkins pipelines to enable execution and analysis of Python and Java unit tests.</paragraph>
 <paragraph index="172" node_type="writer" parent_index="113">Created Terraform code to deploy EKS clusters and related ECR repositories in multiple AWS environments.</paragraph>
 <paragraph index="173" node_type="writer" parent_index="113">Tools: vscode, vim, Jenkins, Docker EE, RHEL, Alpine Linux, BASH, Python, AWS, Terraform, Terraform Enterprise, Bitbucket, git, Python, New Relic, DataDog</paragraph>
 <paragraph index="174" node_type="writer" parent_index="113">Infor Incorporated - Senior DevOps Engineer - (09-2019 - 05-2020)</paragraph>
 <paragraph index="175" node_type="writer" parent_index="113">Automation, Integration, Delivery, and Operations</paragraph>
 <paragraph index="176" node_type="writer" parent_index="113">Designed and implemented GitLab CI pipelines for numerous projects across two teams.</paragraph>
 <paragraph index="177" node_type="writer" parent_index="113">Automated build and deployment of RPMs using Python setuptools, FPM, GitLab CI, Faro and Ansible.</paragraph>

<paragraph index="178" node_type="writer" parent_index="113">Developed and deployed several Docker images for use on GitLab runners.</paragraph>
<paragraph index="179" node_type="writer" parent_index="113">Python Development, Scripting and Software Engineering</paragraph>
<paragraph index="180" node_type="writer" parent_index="113">Spearheaded unit testing efforts using pytest and Pester for testing of Python, BASH, and PowerShell code bases.</paragraph>
<paragraph index="181" node_type="writer" parent_index="113">Added coverage reporting for numerous Python code bases using pytest-cov.</paragraph>
<paragraph index="182" node_type="writer" parent_index="113">Developed BASH and Python implementations of a log monitoring service.</paragraph>
<paragraph index="183" node_type="writer" parent_index="113">Tools: vscode, vim, GitLab, GitLab CI, GitLab Runner, Docker, CentOS, Alpine Linux, BASH, Python (2 and 3), PowerShell, AWS, SumoLogic, Ansible, Faro</paragraph>
<paragraph index="184" node_type="writer" parent_index="113">Disney née Fox Media - Senior DevOps Engineer - (12-2018 - 09-2019)</paragraph>
<paragraph index="185" node_type="writer" parent_index="113">Automation, Integration, Delivery and Operations</paragraph>
<paragraph index="186" node_type="writer" parent_index="113">Improved continuous integration, deployment, and delivery.</paragraph>
<paragraph index="187" node_type="writer" parent_index="113">Developed multiple continuous delivery pipelines with Jenkins.</paragraph>
<paragraph index="188" node_type="writer" parent_index="113">Replicated services and software from Fox Media AWS accounts.</paragraph>
<paragraph index="189" node_type="writer" parent_index="113">Implemented deployment of Docker Swarm clusters to AWS with Terraform. </paragraph>
<paragraph index="190" node_type="writer" parent_index="113">Deployed monitoring packages such as New Relic, Splunk and Sysdig to all systems managed by the operations team with Ansible.</paragraph>
<paragraph index="191" node_type="writer" parent_index="113">Tools: atom, vim, Jenkins, Docker, BASH, CentOS, Alpine Linux Python, PowerShell, JavaScript, New Relic, AWS EC*, AWS CloudWatch, Splunk, Ansible, The Update Framework, Sphinx</paragraph>
<paragraph index="192" node_type="writer" parent_index="113">Infor Incorporated - Senior DevOps Engineer - (06-2018 - 11-2018)</paragraph>
<paragraph index="193" node_type="writer" parent_index="113">Python Development and Software Engineering</paragraph>
<paragraph index="194" node_type="writer" parent_index="113">Updated Step Function and its constituent Lambda functions such that successfully processed events read from a Simple Queue Service queue would be written to a Kinesis stream.</paragraph>
<paragraph index="195" node_type="writer" parent_index="113">Added API key authorization to existing API to enable access via an API Gateway to create the CMDB Client API.</paragraph>
<paragraph index="196" node_type="writer" parent_index="113">Automated the deployment of DynamoDB tables and previously mentioned Kinesis stream with Faro.</paragraph>
<paragraph index="197" node_type="writer" parent_index="113">Wrote Python to acquire statistics and publish them to a Grafana endpoint.</paragraph>
<paragraph index="198" node_type="writer" parent_index="113">Implemented GitLab Continuous Integration for testing and deployment.</paragraph>
<paragraph index="199" node_type="writer" parent_index="113">Gahan Corporation - Director of Technology - (05-2016 - 12-2023)</paragraph>
<paragraph index="200" node_type="writer" parent_index="113">Technology</paragraph>
<paragraph index="201" node_type="writer" parent_index="113">Managed gahan-corporation.com mail services using Postfix with Courier (IMAP) then migrating to Dovecot with Ansible, Docker, AWS ECS, and AWS Route53.</paragraph>
<paragraph index="202" node_type="writer" parent_index="113">An SQL query walks into a bar, walks up to two tables and says, ‘do you mind if I join

you?</paragraph>
 <paragraph index="203" node_type="writer" parent_index="113">Clients</paragraph>
 <paragraph index="204" node_type="writer" parent_index="113">CSI - Principal DevOps Engineer / Principal Python Engineer</paragraph>
 <paragraph index="205" node_type="writer" parent_index="113">Automated management of development, staging, and production Odoo environments, as well as PostgreSQL servers and databases with Ansible and Docker.</paragraph>
 <paragraph index="206" node_type="writer" parent_index="113">Designed and deployed CI/CD system with Jenkins on Linode which enabled automated testing of Python and PostgreSQL with PyTest, Ansible, and Docker.</paragraph>
 <paragraph index="207" node_type="writer" parent_index="113">Developed Odoo Enterprise Modules with Python, PostgreSQL, and SQLAlchemy that required extreme reduction of cyclomatic complexity, complete rewrites to comply with PEP8, in addition to a far greater number of bug fixes than was ever accounted for.</paragraph>
 <paragraph index="208" node_type="writer" parent_index="113">Discovered mission critical inconsistencies in accounting and inventory data which was causing valuation errors more than one million dollars using Odoo, PostgreSQL, Python, and SQLAlchemy.</paragraph>
 <paragraph index="209" node_type="writer" parent_index="113">Tools: vim, BASH, AWS EC2, AWS S3, NGINX, Jenkins, Ansible, Docker, HTML, CSS, PostgreSQL, Sphinx, Odoo ERP, GitHub, Linode</paragraph>
 <paragraph index="210" node_type="writer" parent_index="113">Abiogenix, Incorporated - Principal DevOps Engineer</paragraph>
 <paragraph index="211" node_type="writer" parent_index="113">Deployed, managed, and automated configuration of Atlassian suite (JIRA, Bitbucket, Confluence) with Amazon EC2, Amazon ECS, Ansible, Docker, and Ubuntu.</paragraph>
 <paragraph index="212" node_type="writer" parent_index="113">Deployed and maintained mail services for my-ubox.com with, Postfix Courier (IMAP), Ansible, Amazon ECS, and Docker.</paragraph>
 <paragraph index="213" node_type="writer" parent_index="113">Administered and automated maintenance of Odoo ERP servers (development and production) with Ansible, Python, Docker, Amazon ECS and PostgreSQL.</paragraph>
 <paragraph index="214" node_type="writer" parent_index="113">Trained team members on proper use of git and gitflow to manage branching and deployment to production.</paragraph>
 <paragraph index="215" node_type="writer" parent_index="113">Implemented push-to-deploy on Abiogenix web repositories to allow developers to deploy to production without using ssh.</paragraph>
 <paragraph index="216" node_type="writer" parent_index="113">Tools: vim, BASH, AWS EC2, S3, Route53, IAM, ECS, Ansible, Docker, Stripe, Python, Django, git, Bitbucket, JIRA, Confluence, Tomcat, Apache2</paragraph>
 <paragraph index="218" node_type="writer" parent_index="113">Network of One - DevOps Engineer</paragraph>
 <paragraph index="219" node_type="writer" parent_index="113">Configured, deployed, and provided instruction for a CRM system using Ansible, BASH, and AWS EC2.</paragraph>
 <paragraph index="220" node_type="writer" parent_index="113">Developed automated distribution of public keys to EC2 instances based on user groups with AWS EC2, AWS IAM, Python, BASH and Ansible.</paragraph>
 <paragraph index="221" node_type="writer" parent_index="113">Automated configuration of EC2 instances with Ansible and BASH.</paragraph>
 <paragraph index="222" node_type="writer" parent_index="113">Tools: vim, BASH, Ansible, Python, AWS EC2, AWS IAM, NGINX, Docker, IBM DB2 AWS RDS, PostgreSQL</paragraph>
 <paragraph index="223" node_type="writer" parent_index="113">CAA - DevOps Engineer / Lead Python Developer (09-2015 - 04-2016)</paragraph>
 <paragraph index="224" node_type="writer" parent_index="113">Automation, Integration, Delivery and Operations</paragraph>

<paragraph index="225" node_type="writer" parent_index="113">Implemented CI/CD for an internal project which was a year behind schedule to enable the developers to deliver bug fixes and new features daily with Jenkins, Ansible, Python, Web2Py, and Ansible Tower.</paragraph>
<paragraph index="226" node_type="writer" parent_index="113">Managed and monitored cloud infrastructure of roughly 30 EC2 instances using Ansible, Ansible Tower, BASH, CentOS, Ubuntu, and New Relic.</paragraph>
<paragraph index="227" node_type="writer" parent_index="113">Alerted stakeholders to failing jobs in Ansible Tower or non-functional infrastructure with New Relic, PagerDuty, Python, and Ansible.</paragraph>
<paragraph index="228" node_type="writer" parent_index="113">Python Development and Software Engineering</paragraph>
<paragraph index="229" node_type="writer" parent_index="113">Trained and guided WorkDay migration team in the use of Python, Pandas and JupyterHub through completion of the project.</paragraph>
<paragraph index="230" node_type="writer" parent_index="113">Tools: vim, BASH, Ansible, Ansible Tower, Python, AWS EC2, NGINX, uWSGI, Python, Web2Py, Jenkins, Docker, Ubuntu, CentOS, JupyterHub, git, GitHub, MySQL, Slack</paragraph>
<paragraph index="231" node_type="writer" parent_index="113">Toyota R&D - DevOps Engineer / Lead Python Developer - (11-2014 - 07-2015)</paragraph>
<paragraph index="232" node_type="writer" parent_index="113">Automation, Integration, Delivery and Operations</paragraph>
<paragraph index="233" node_type="writer" parent_index="113">Automated deployment of Customer 360 services to reduce time to deploy from several weeks to less than an hour with Ansible, BASH, Java, Ant, Maven, Tomcat, and Scala Build Tool.</paragraph>
<paragraph index="234" node_type="writer" parent_index="113">Managed configuration and maintenance of Customer 360 cloud infrastructure consisting of roughly 10 Red Hat Linux EC2 instances using Ansible, BASH, and Python.</paragraph>
<paragraph index="235" node_type="writer" parent_index="113">Designed, deployed, and implemented automated tests and deployment of Customer 360 code written in Java, Scala, and Python with Ansible, BASH, Ant, Maven, and Jenkins.</paragraph>
<paragraph index="236" node_type="writer" parent_index="113">Supported a team of 25 developers with Jenkins CI, Bitbucket, and Ansible to ensure they were able to release on time.</paragraph>
<paragraph index="237" node_type="writer" parent_index="113">Python Development and Software Engineering</paragraph>
<paragraph index="238" node_type="writer" parent_index="113">Provided instruction on the use of git, gitflow, Bitbucket, and best practices for commits and merges to Customer 360 developers after gaining consensus and approval of Customer 360 management with an A3 presentation.</paragraph>
<paragraph index="239" node_type="writer" parent_index="113">Refactored and developed web scraping software using ScraPY, Python, and Ansible.</paragraph>
<paragraph index="240" node_type="writer" parent_index="113">Provided instruction on compliance to team of remote Python developers.</paragraph>
<paragraph index="241" node_type="writer" parent_index="113">Participated in sprint planning and review cycle with Customer 360 team of 25 developers.</paragraph>
<paragraph index="242" node_type="writer" parent_index="113">Tools: vim, BASH, Ansible, Jenkins, Python, Java, Ant, Maven, Scala, Tomcat, SBT, git, AWS EC2, CentOS, Bitbucket, JIRA, REST, SCRUM, gitflow, ScraPY</paragraph>
</indexing>