

# PrepAwayExam

PrepAwayExam

> Contact Us

Login / Register

Search...



HOME

ALL VENDORS

★ GUARANTEE

? FAQ

TESTIMONIALS

CART (0)

## Pass Your Next Certification Exam Fast!

Everything you need to prepare, learn & pass your certification exam easily.

365 days free updates. First attempt guaranteed success.

Try **Online Engine** before you buy

### Instant Download



After Payment, our system will send you the products you purchase in mailbox in a minute after payment. If not received within 2 hours, please contact us.

### 365 Days Free Updates



Free update is available within 365 days after your purchase. After 365 days, you will get 50% discounts for updating.



### Money Back Guarantee

Full refund if you fail the corresponding exam in 60 days after purchasing. And Free get any another product.



### Security & Privacy

We respect customer privacy. We use McAfee's security service to provide you with utmost security for your personal information & peace of mind.

<http://www.prepawayexam.com/>

High-efficient Exam Materials are the best high pass-rate Exam Dumps

**Exam** : **Development-Lifecycle-and-Deployment-Architect-JPN**

**Title** : Salesforce Certified  
Development Lifecycle and  
Deployment Architect  
(Development-Lifecycle-  
and-Deployment-  
Architect日本語版)

**Vendor** : Salesforce

**Version** : DEMO

### QUESTION NO: 1

Universal Containers (UC) は、EMEA、AMER、APAC で事業を展開するエンタープライズ金融会社です。

規制要件のため、UC には地域ごとに個別の Salesforce

組織があります。各組織にはリージョンのニーズに合わせた独自のカスタマイズがありますが、すべてのリージョンの要件に適用される標準プロセスもあります。

デプロイメントアーキテクトとして、複数組織のデプロイメント戦略について何を考慮する必要がありますか？

- A. 管理パッケージを使用してメタデータを本番組織にリリースします。
- B. 非管理パッケージを使用してメタデータを本番組織にデプロイします。
- C. パッケージ開発モデルを使用してメタデータを本番組織にデプロイします。
- D. 変更セットを使用してメタデータを本番組織にリリースします。

**Answer: C**

Explanation:

Deploying metadata to production orgs using package development model is the best option for the multi-org deployment strategy, as it allows you to create modular and reusable packages that can be easily installed and updated across different orgs. Deploying metadata to production orgs using managed packages is not suitable for this scenario, as managed packages are typically used by ISVs to distribute their applications to customers, and they have some limitations and restrictions that may not fit the requirements of UC. Deploying metadata to production orgs using unmanaged packages is also not a good option, as unmanaged packages are mainly used for one-time distribution of components, and they do not support upgrades or dependencies. Deploying metadata to production orgs using change sets is not feasible for this scenario, as change sets can only be used to deploy metadata between connected orgs in the same Salesforce instance, and UC has separate orgs for each region. See [Package Development Model] for more details.

### QUESTION NO: 2

Universal Containers (UC) は、3,000 ユーザーのコンタクト センターに Service Cloud を導入しています。彼らは持っている

約1,000万人の顧客。予想される平均速度応答時間は、1,500 人の同時ユーザーで 5 秒未満です。UC

がページの応答時間を測定するには、どのような種類のテストが役立ちますか？

- A. 単体テスト。
- B. 負荷テスト。
- C. システム統合テスト。
- D. ストレス テスト。

**Answer: B**

Explanation:

Load testing is the type of testing that will help UC measure the page response time. Load testing simulates the expected number of concurrent users and measures how the system performs under normal load conditions. Unit testing, system integration testing, and stress testing are not designed to measure the page response time.

### QUESTION NO: 3

Universal Containers (UC) は最近、独自の Salesforce 組織を持つ他の企業を買収しました。これらの会社は新しい UC ビジネスユニットとして統合されました。

CEO はアーキテクトに、次の 2

つの主要要素を考慮して組織戦略を見直すよう依頼しました。

\* CEO は、すべてのビジネス ユニット間でのビジネス プロセスの標準化を望んでいます。

\*

異なるビジネスユニットは異なる顧客と専門知識を持っているため、ビジネスプロセスの統合は必要ありません。

このシナリオでアーキテクトはどの組織戦略を推奨する必要がありますか?またその理由は何ですか?

A. 単一組織戦略。高レベルのビジネス

プロセス標準化を単一組織で実装する方が簡単です。

B. 複数組織戦略。多様化したビジネス ユニットが他のビジネス

ユニットと同じスペースで作業することに慣れるのは珍しいためです。

C. 複数組織戦略。共通の管理パッケージをさまざまな事業単位の組織にデプロイできます。

D. 単一組織戦略。組織の数が増えるとコストが増加します。

**Answer: A**

Explanation:

A single-org strategy is the best option for UC, as it will enable them to achieve business process standardization among all business units, which is the main goal of the CEO. A multi-org strategy would make it harder to enforce consistent processes and policies across the different business units, and would also increase the costs and complexity of managing multiple orgs. A common managed package could help with some aspects of standardization, but it would not cover all the possible scenarios and customizations that UC might need.

### QUESTION NO: 4

Universal Containers には 5

つの開発チームがあります。チームのパフォーマンスは良好ですが、バグの数は増加しています。各スプリントの後、コードを理解して変更を加えるにはさらに時間がかかります。

パフォーマンスを向上させる 2 つの方法は何ですか?

2 つの答えを選択してください

A. すべての変更を分析/承認するチームを定義します。

B. コード標準を定義して準拠します。

C. スプリント レビュー プロセス。

D. バグの生成者を特定するためのバージョン管理システム。

**Answer: B C**

Explanation:

To improve the performance of the development teams, the following ways can be suggested: Define and follow code standards, and implement a sprint review process. Code standards can help ensure consistency, readability, and maintainability of the code, as well as reduce errors and bugs. A sprint review process can help evaluate the work done in each sprint, demonstrate the functionality, and gather feedback from the stakeholders and users.

### QUESTION NO: 5

販売およびサービス製品は、第 2 世代の管理パッケージを使用する 2 つのチームによって作成されます。

セールス

チームはサービス製品の特定の機能を使用しますが、アーキテクトは、このチームがサービス

チームによって公開された機能のみを使用するようにしたいと考えています。他のチームはこれらと同じ機能を使用しません。

建築家は何を推奨すべきでしょうか？

A. 同じ名前空間を持つ 2 つの第 2 世代管理パッケージを作成し、@namespaceAccessible アノテーションと共有する必要があるメソッドを設定します。

B. Sales 名前空間と service 名前空間を持つ 2 つの管理パッケージを作成します。salesAccessible アノテーションと共有するメソッドを設定する

C. 両方の製品を含む管理パッケージを作成し、各チームの承認者によるコード レビュー プロセスを作成します。

D. 2 つの管理パッケージを作成します。Sales ユーザーが公開された関数を呼び出す権限を持っている場合にトークンを返す認証関数を Service パッケージ内に作成します。サービス関数でトークンを検証します。

**Answer: A**

Explanation:

The architect should recommend creating two second generation managed packages with the same namespace and setting the methods that should be shared with the @namespaceAccessible annotation. This will allow the Sales team to access the specific functions of the Service product without exposing them to other teams or customers. Creating two managed packages with different namespaces will not allow the Sales team to access the Service functions, unless they are declared as global, which will expose them to everyone. Creating a managed package with both products will not allow the separation of the products and the control of the functions. Creating an authentication function in the Service package will add unnecessary complexity and overhead to the solution.

### QUESTION NO: 6

Universal Containers

は、複数のプロジェクトに取り組むチームを開発しました。彼らは、コード/構成を追跡および管理するためのソース管理ツールを検討しています。ソース管理ツールが提供する 2 つの利点は何ですか？ 2つ選択してください

A. 分散したチームが孤立して作業できる機能を提供します。

B. 自動化されたコード/構成のデプロイメントを提供します。

C. コード/構成の変更をバックアップする機能を提供します。

D. コード/構成の問題を自動的に特定する機能を提供します。

**Answer: A C**

Explanation:

A source control tool provides the ability for distributed teams to work in isolation, as they can create branches and merge their changes later. It also provides the ability to backup

code/configuration changes, as they can be stored in a remote repository and retrieved if needed.

**QUESTION NO: 7**

アプリケーション

ライフサイクル手法をサポートするためにアーキテクトが推奨するツールを 3 つ選択してください 3 つの回答を選択してください

- A. データベース管理システム
- B. バージョン管理リポジトリ
- C. ミドルウェア
- D. 継続的統合ツール
- E. 問題追跡ツール

**Answer:** B D E

Explanation:

To support application lifecycle methodology, you need tools that can help you manage the source code, automate the deployment process, and track the issues and bugs. A version control repository is a tool that allows you to store, track, and collaborate on the source code of your application. A continuous integration tool is a tool that allows you to automate the deployment of your code to different environments, as well as run tests and validations. An issue tracking tool is a tool that allows you to record, monitor, and resolve the issues and bugs that arise during the development and testing phases. A database management system is a tool that allows you to store, manipulate, and query data, but it is not directly related to application lifecycle methodology. A middleware is a software layer that facilitates communication and data exchange between different applications, but it is not directly related to application lifecycle methodology either.

**QUESTION NO: 8**

アーキテクトはユニバーサル コンテナ (UC)

プロジェクトに取り組んでいますが、セキュリティ上の懸念により、UC セキュリティ チームはアーキテクトに運用アクセスを提供できません。代わりに、中央のリリース管理 チームが、すべての開発チームに対して実稼働環境のデプロイメントを実行する責任を負いま す。

アーキテクトはメタデータ API

をどのように活用して、プロジェクトの機能をデプロイするために必要なメタデータ コンポーネントがリリース管理チームに適切に伝達されていることを確認する必要がありますか？

- A. すべてのコンポーネントのスプレッドシートを提供し、メタデータ API の読み取り Metadata() 呼び出しを利用します。
- B. ロック解除されたパッケージのバージョンをリリース管理チームに通知します。
- C. 各サンドボックスに変更セットを作成し、リリース管理チーム用の package.xml ファイルをダウンロードします。
- D.

D.

デプロイ元となるサンドボックスからの監査証跡のコピーをリリース管理チームに提供しま す。



release, as it is not necessary to upgrade the SOAP integrations to the newest WSDL, unless they want to use the new features or fields introduced by the upgrade. The SOAP integrations will continue to work with the previous WSDL versions, as they are backward compatible.

#### QUESTION NO: 10

Universal Containers (UC) はガバナンス

フレームワークを実装しており、アーキテクトにリリース計画に関する推奨事項を作成するよう依頼しました。アーキテクトがリリースを計画する際に行うべき 2 つの決定はどれですか? 2 つの答えを選択してください

- A. 既存の機能をテストして、回帰が発生しないことを確認する方法。
- B. Salesforce が UC リリースが完了するまでポッドのアップグレードを待機するかどうか。
- C. 問題がある場合に、以前の Salesforce リリースにロールバックする方法。
- D. 問題がある場合に、新しい UC 機能リリースをテストする時期。

**Answer:** A D

Explanation:

How to test existing functionality to ensure no regressions are introduced is a decision that the Architect should make when planning for releases, as it is part of the quality assurance process and helps to ensure that the new changes do not break the existing functionality. When to test a new UC feature release is also a decision that the Architect should make when planning for releases, as it is part of the release schedule and helps to coordinate the testing activities with the development and deployment activities. Whether Salesforce will wait to upgrade the pod until after a UC release is complete is not a decision that the Architect can make, as it is determined by Salesforce and depends on the release window and the pod assignment. How to roll back to the previous Salesforce release if there are issues is not a decision that the Architect can make, as it is not possible to roll back to a previous Salesforce release once the upgrade is done.

#### QUESTION NO: 11

Universal Containers は、レガシー

システムへの新しい複雑な統合を構築しています。レガシー

システムも大幅なアップグレードが行われます。上級指導部は、統合されたプログラムが予定通り完了することを取締役会に約束しました。この計画にはどのようなリスクがありますか?

- A. 期限は Salesforce リリース中にスケジュールされています
- B. プロジェクト チームはウォーターフォール手法を使用することを決定しました。
- C. レガシー システム チームはアジャイル手法を使用しています
- D. 依存関係のある複数の作業ストリームは、稼働開始に影響を与える可能性があります

**Answer:** D

Explanation:

D is the correct answer, as the risk with this plan is that multiple work-streams with dependencies could impact the go-live. If the legacy system upgrade and the new integration are not aligned and coordinated, there could be delays, errors, or failures in the project delivery. A is incorrect, as the deadline being scheduled during a Salesforce release is not a risk, but an opportunity to leverage the new features and enhancements that Salesforce

provides. B is incorrect, as the project team using the Waterfall methodology is not a risk, but a choice that depends on the project scope, complexity, and requirements. C is incorrect, as the legacy system team using an Agile methodology is not a risk, but a choice that depends on the project scope, complexity, and requirements. You can learn more about the risk management in the Project Management Strategies for Salesforce Implementations module on Trailhead.

### QUESTION NO: 12

Universal Containers は、Salesforce

の地域の実稼働インスタンスを管理するグローバル組織です。ある地域では、輸送コンテナを追跡するための新しいアプリケーションを作成しました。

CIO は、この新しいアプリケーションがすべての Salesforce

インスタンスによってグローバルに使用され、さらにローカル管理者によって地域ごとに保守および変更されるように要求しました。

このリクエストをサポートする 2 つの展開ツールはどれですか？

2 つの答えを選択してください

- A. 変更セット B
- B. 開発者コンソール
- C. ANT 移行ツール
- D. Salesforce 拡張機能を備えた VS コード

**Answer:** C D

Explanation:

The two deployment tools that will support the request are the ANT Migration Tool and VS Code with Salesforce Extension. These tools allow the developers to deploy metadata components from one Salesforce org to another, and also to maintain and modify the code locally in their own machines. Change Sets and Developer Console are not suitable for this scenario, because they do not support deploying to multiple orgs or working offline.

### QUESTION NO: 13

Universal Containers (UC) はエンタープライズ Salesforce

変革の取り組みに着手しており、UC

は構築段階でさまざまなサンドボックスへの展開を合理化し、自動化したいと考えています。UAT

が顧客に受け入れられると、同社は実稼働環境の展開も自動化するよう要求しました。

導入アーキテクトとして、顧客の要件を満たすための推奨事項は何ですか？

- A. Continues 統合および Continues デプロイメント  
ツールを使用し、サンドボックスと運用環境にデプロイするためのパイプラインを構築することをお勧めします。
- B. SFDX の使用を推奨し、環境ごとに実行する手順を含むデプロイメント  
コマンドを文書化します。
- C. ANT スクリプトを使用し、スクリプトを実行するカスタム  
アプリケーションを構築し、変更セットを使用してサポートされているメタデータをデプロイすることをお勧めします。
- D. デプロイメントコンポーネントをパッケージ化した AppExchange  
ソリューションを使用することをお勧めします。デプロイメントウィザードを実行してデプ

ロイメント結果を追跡できます。

**Answer: A**

Explanation:

The recommendation to satisfy the customer requirements is to use a Continuous Integration and Continuous Deployment tool and build the pipeline to deploy to sandboxes and production. A Continuous Integration and Continuous Deployment tool can automate the process of building, testing, and deploying the changes to different environments, as well as provide feedback and visibility into the deployment status and results. This can help to streamline and accelerate the deployment process, as well as to ensure consistency and quality across the environments. Using SFDX and documenting the deployment commands with steps to be executed for each environment is not a good recommendation, as it still requires manual intervention and execution, which can be error-prone and time-consuming. Using ANT script and building a custom application to run the script and use change sets to deploy supported metadata is not a good recommendation, as it involves using multiple tools and methods, which can increase the complexity and risk of the deployment process. Using an AppExchange solution that packages the deployment components and you can run the deployment wizard to track deployment result is not a good recommendation, as it may not support all the metadata types and features that need to be deployed, and it may not integrate well with the SFDX tools and methodologies that UC is using.

#### QUESTION NO: 14

Universal Containers

には、本番環境で見つかったバグを分析して修正するために使用される完全なサンドボックスがあります。

運用環境で見つかったバグをこの完全なサンドボックスでより簡単に分析できるようにするために、アーキテクトは次の2つの項目を推奨する必要がありますか？

2つの答えを選択してください

**A.** 運用環境でデプロイするたびに完全なサンドボックスを更新します。

**B.**

本番環境の新しいデータと変更されたデータを完全なサンドボックスにコピーする毎日のプロセスを作成します。

**C.**

運用環境にデプロイする前に、同じプロセスをこのサンドボックスで実行する必要があります。

**D.** 完全なサンドボックスでデータの更新を実行します。

**Answer: A C**

Explanation:

To ensure that bugs found in production are more easily analyzed in the full sandbox, the architect should recommend to refresh the full sandbox after every deployment in production, and to perform the same process in the sandbox before any deployment in production. This way, the full sandbox will be in sync with the production org and will have the same metadata and code. Creating a daily process of copying new and changed data in production to the full sandbox is not necessary, as the full sandbox already has a copy of the production data. Performing a Refresh Data in the full sandbox is not possible, as this option is only available for partial copy and developer pro sandboxes.

### QUESTION NO: 15

パッケージ開発モデルの 3 つの利点とは何ですか？

3 つの答えを選択してください

A. チームの開発とコラボレーションを改善します。

B.

変更セットを使用する必要性を排除します。変更セットは、パッケージ開発モデルでの作業が面倒になる可能性があるため、使用すべきではありません。

C. 自動テストと継続的統合を促進します。

D. 変更を手動で追跡する必要性が大幅に減少します。

E. 独自のソース管理を提供するため、ソースを任意の Sandbox 組織にデプロイできます。

**Answer:** A C D

Explanation:

The advantages of the package development model are improving team development and collaboration, facilitating automated testing and continuous integration, and significantly reducing the need for manually tracking changes. The package development model allows the developers to work on modular and reusable components that can be easily tested and deployed. The package development model does not eliminate the need of using change sets, as they can still be used for deploying non-packaged components or metadata. The package development model does not provide its own source control, but rather relies on external source control systems such as Git.

### QUESTION NO: 16

Universal Containers (UC) は、変更セット開発モデルを使用して、顧客サービスチームが使用できるように Salesforce Service Cloud の拡張機能を開発しました。

最近、UC は AppExchange

アプリの拡張機能を開発する会社を買収しました。買収された会社の開発チームは組織開発モデルを使用します。Universal Containers の CTO

は、両方のチームが単一の組織で作業し、同じ一連のプロセスに従うことを望んでいます。統合開発チームが使用するためにアーキテクトはどの開発モデルを推奨する必要がありますか？

A.

組織開発モデル。買収した企業のチームがすでに使用しており、変更セット開発モデルよりも優れているためです。

B. パッケージ開発モデル。コードを記述せずに、宣言型 (ポイント アンド クリック) 開発ツールを使用してパッケージを作成およびデプロイできるためです。

C.

パッケージ開発モデル。チームは、他のプロジェクトのアーティファクトから独立してテストおよびリリースできるリリース アーティファクトを構築できます。

D. 変更セット開発モデル。UC はすでにそれを使用しているため、抵抗は少なくなります。

**Answer:** C

Explanation:

The development model that the architect should recommend to be used by the consolidated development team is the package development model. This model allows teams to build

release artifacts that can be tested and released independently from artifacts for other projects, using unlocked packages or second-generation managed packages. This model can help improve the modularity, reusability, and maintainability of the code and configuration, as well as enable source-driven development and continuous integration and delivery. The org development model is not a good choice for the consolidated development team, as it is based on working directly in an org and using change sets or metadata API tools to deploy changes. This model can lead to conflicts, dependencies, and governance issues, especially when working on multiple projects or with multiple teams. The change set development model is also not a good choice for the consolidated development team, as it is based on using change sets to deploy changes between connected orgs. This model can be slow, error-prone, and limited, as it does not support all metadata types, dependencies, or automation. The package development model does not allow packages to be created and deployed using declarative (point-and-click) development tools, without writing code. This is a wrong statement, as the package development model supports both declarative and programmatic development tools, and requires writing code to create and install packages using the Salesforce CLI or APIs. See Package Development Model for more details.

#### QUESTION NO: 17

Universal Containers (UC) は、AppExchange を対象とした管理パッケージを開発しました。製品には、レイアウトをカスタマイズおよび作成するためのいくつかの Apex コードが含まれています。UC はパッケージのテスト段階にあるため、まだ認定されていません。対象組織でのテスト中に、レイアウトの Apex コードが失敗します。

テスト中に Apex

クラスがターゲット組織のメタデータにアクセスできないのはなぜですか？

- A. メタデータへのアクセスを許可する Apex 設定がオンになっていません。
- B. UC はカスタム メタデータ タイプ内で Apex 設定を有効にする必要があります。
- C. ソリューションには欠陥があります。UC は、Web サービス呼び出しからツール API を利用してレイアウトを変更する必要があります。
- D. UC は、Salesforce セキュリティ レビューによって管理パッケージの認定を取得する必要があります。

**Answer:** D

Explanation:

The reason why the Apex classes are not able to access the metadata of the target org during testing is that UC needs to get the managed package certified by the Salesforce security review. This is because Apex code in a managed package can only access the metadata of the target org if the package has passed the security review and has been granted the Modify Metadata permission. See Apex Metadata API for more details.

#### QUESTION NO: 18

Universal Containers (UC)

はさまざまなパートナーと連携しており、日常的な展開タスクを処理する管理リソースはほとんどありません。その結果、UC はメタデータ API を使用して展開を自動化する方法を見つけたいと考えています。メタデータ API ベースのデプロイメントを使用する場合、メタデータ API の 2

つの制限事項を考慮する必要があるのはどれですか? 2 つの答えを選択してください

- A. 最大 10,000 個のファイルをデプロイしますが、10,000 個を超えるファイルを取得します。
- B. デプロイされた .zip ファイルの最大サイズは 400MB です。
- C. デプロイされた .zip ファイルの最大サイズは 39MB です。
- D. 一度に最大 10,000 個のファイルを展開および取得します。

**Answer:** A C

Explanation:

The maximum size of deployed .zip file is 39MB for Metadata API. You can deploy up to 10,000 files, but retrieve more than 10,000 files using Metadata API.

### QUESTION NO: 19

Universal Containers

では、複数のプロジェクトが並行して開発されています。プロジェクトの 1

つはテスト段階にあり、テスト

チームは本番環境にデプロイされる項目に関する問題のリストを発見しました。プロジェクトの期限が短いため、顧客チームはテスト

サンドボックスで修正を行ってから運用環境にデプロイすることを提案しました。建築家は何を推奨すべきでしょうか?

- A. テスト環境の問題を修正し、実稼働環境にデプロイするための顧客チームの提案を推奨します。
- B. 開発環境で問題を修正し、変更を実稼働環境にデプロイすることを推奨します。
- C. 開発サンドボックスで問題を修正し、テストに移行し、テスト後に実稼働環境にデプロイすることを推奨します。
- D. テスト環境で問題を修正し、変更を開発サンドボックスに移行することをお勧めします。

**Answer:** C

Explanation:

The best practice is to fix the issues in the development sandbox, migrate them to the testing sandbox, and deploy to production after testing. This ensures that the development and testing environments are in sync and that the changes are properly tested before going to production. Fixing the issues in the testing environment and deploying them to production may introduce errors or conflicts, as the development environment may not have the same changes. Fixing the issues in the test environment and migrating them to the development sandbox may also cause errors or conflicts, as the development environment may have other changes that are not ready for testing.

### QUESTION NO: 20

Universal Containers は、パートナー

コミュニティを実装するプロジェクトを開始したところです。このアプリケーションは、大規模な Salesforce ユーザー

ベースが現在使用している本番環境にデプロイされます。プロジェクト

マネージャーは、開発チームとテスト

チームが単一の開発者サンドボックスを使用することを主張しました。このアプローチにはどのようなリスクがありますか？

A.

開発者サンドボックスの容量制限により、テスターはプラットフォームの制限に遭遇します。

B. テスターは、分離フォームの開発がないため、スループットテストの機能変更が発生します。

C.

開発者サンドボックスに大量のユーザーが存在するため、テスターはガバナ制限に達します。

D. 開発者サンドボックスの更新にはかなりの時間がかかります。

**Answer: B**

Explanation:

Testers will experience functional changes throughout testing due to not having isolation from development.

Using a single developer sandbox for both development and testing is not a good practice, as it does not provide a stable and consistent environment for testing. The developers may make changes to the code or configuration that affect the functionality or behavior of the application, which may cause the testers to encounter unexpected results or errors.

#### QUESTION NO: 21

Universal Containers (UC) は、リリース管理プロセスの一環として Salesforce 本番環境の更新を検討しています。UC が運用環境の展開に関して考慮すべき 3 つのベストプラクティスはどれですか？ 3 つお選びください。

A. メンテナンス期間を事前に発表します。

B. ロールバック戦略を定義します。

C. 導入日にすべてのユーザーに通知します。

D. Salesforce アップグレードを伴うリリースをスケジュールします。

E. 本番環境での構成変更を一時的に中断します。

**Answer: A B E**

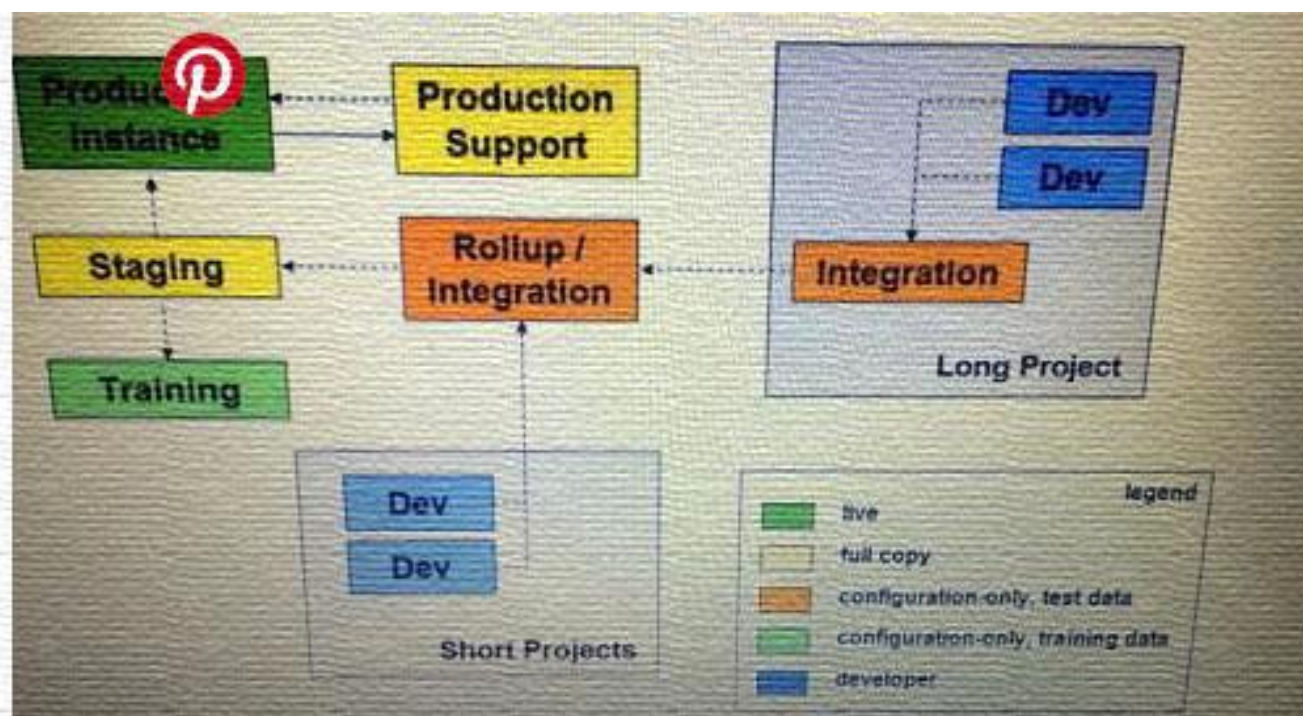
Explanation:

Announcing the maintenance window ahead of time, defining a rollback strategy, and temporarily suspending configuration changes in production are all best practices for production deployment. These practices help to minimize the risk of errors, downtime, and conflicts during the deployment process.

#### QUESTION NO: 22

このリリースは、Salesforce が本番環境を更新してから 1 週間後の週末 (冬から春など) にデプロイされます。UC

は、サンドボックスの完全な更新には数日かかる場合があることを発見しました。アーキテクトは最適な導入計画として何を提案すべきでしょうか？



**A. 本番稼働の 2**

週間前に、ステージングにデプロイしてから、ステージングおよび運用サポートサンドボックスを更新します。本番稼働時にステージングから本番環境にデプロイする

**B. 公開の約 6 週間前に、サンドボックスがリリース**

プレビューに含まれていることを確認してください。本番稼働の 1

週間前に、ステージングにデプロイします。本番稼働時にステージングから本番環境にデプロイする

**C. ライブ開始の 1**

か月前に、ステージングと実稼働サポートにデプロイします。運用開始時に運用サポートから運用にデプロイする

**D. 本番稼働の 1 週間前に、ステージング**

サンドボックスの更新を開始し、すぐにステージングにデプロイします。本番稼働時にステージングから本番環境にデプロイします。

**Answer: B**

Explanation:

The best option is to ensure the sandbox will be on the release preview, which means it will be upgraded to the new platform release before the production environment. This will allow the team to test the deployment in a realistic scenario and catch any potential issues. Option A is not ideal, as the staging and production support sandboxes will not be on the same platform version as the production environment. Option C is also not ideal, as the production support sandbox will not be on the same platform version as the production environment. Option D is risky, as the staging sandbox refresh may not complete in time for the go-live.

**QUESTION NO: 23**

Universal Containers

は、本番環境に毎週リリースしたいと考えています。バグが誤って実稼働環境に持ち込まれるリスクを軽減するには、どのようなアプローチがありますか？

**A. ユーザー受け入れテスト**

